

PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2004-038941

(43)Date of publication of application : 05.02.2004

(51)Int.Cl. G06F 12/00
G06F 13/00
G10K 15/02

(21)Application number : 2003-120259 (71)Applicant : MATSUSHITA ELECTRIC IND CO LTD

(22)Date of filing : 24.04.2003 (72)Inventor : SHEN MEI SHEN
HUANG ZHONGYANG
JI MING
SENOO TAKANORI

(30)Priority

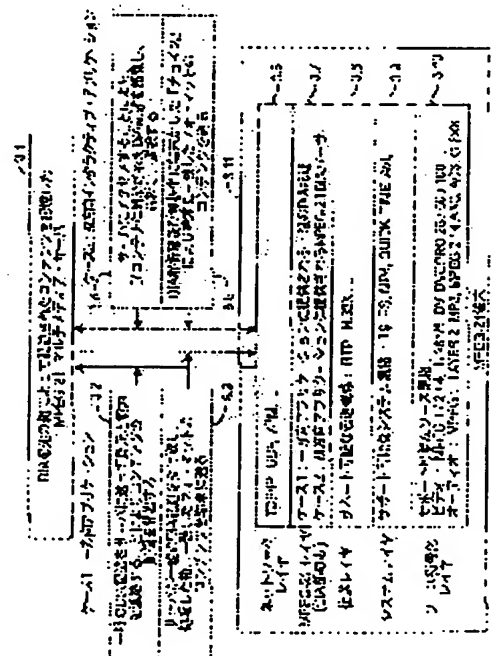
Priority number : 2002127160 Priority date : 26.04.2002 Priority country : JP

(54) CONTENT ADAPTATION METHOD FOR TERMINAL DEVICE, SERVER AND GATEWAY OF UNIVERSAL MULTIMEDIA FRAMEWORK

(57)Abstract:

PROBLEM TO BE SOLVED: To provide a system of a universal multimedia framework terminal that performs content adaptation.

SOLUTION: The system performs content adaptation including a step of building a terminal with content decoding tools compliant with given standards, a step of describing the terminal with DIA (digital item adaptation) descriptions, a step of attaching the DIA descriptions to the terminal, a step of sending and submitting the DIA descriptions to a server for retrieving a piece of content, whereby the server parses and processes the DIA descriptions, a step of selecting content matched in format with the DIA descriptions submitted by the terminal, and a step of delivering the content to the terminal.



(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2004-38941

(P2004-38941A)

(43) 公開日 平成16年2月5日(2004. 2. 5)

(51) Int. Cl. 7

F I

G 0 6 F 12/00

G 0 6 F 12/00

5 4 6 R

テーマコード (参考)

G 0 6 F 13/00

G 0 6 F 13/00

5 4 0 A

5 B 0 8 2

G 1 0 K 15/02

G 1 0 K 15/02

審査請求 未請求 請求項の数 15 O L 外国語出願 (全 130 頁)

(21) 出願番号	特願2003-120259 (P2003-120259)	(71) 出願人	000005821
(22) 出願日	平成15年4月24日 (2003. 4. 24)		松下電器産業株式会社
(31) 優先権主張番号	特願2002-127160 (P2002-127160)	(74) 代理人	大阪府門真市大字門真1006番地
(32) 優先日	平成14年4月26日 (2002. 4. 26)		100086405
(33) 優先権主張国	日本国 (JP)		弁理士 河宮 治
		(74) 代理人	100098280
			弁理士 石野 正弘
		(72) 発明者	シェン メイ・シェン
			シンガポール534415シンガポール、
			タイ・セン・アベニュー、ブロック102
			2、04-3530番、タイ・セン・イン
			ダストリアル・エステイト、パナソニック
			・シンガポール研究所株式会社内

最終頁に続く

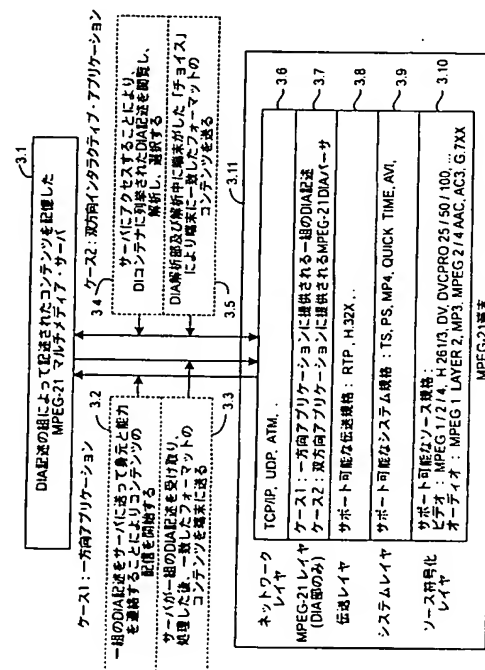
(54) 【発明の名称】 ユニバーサル・マルチメディア・フレームワークの端末装置、サーバ、及びゲートウェイのコンテンツ適合方法

(57) 【要約】

【課題】 コンテンツ適合を行うユニバーサル・マルチメディア・フレームワーク端末の装置を提供する。

【解決手段】 本装置は、ある一定の規格に準拠するコンテンツ復号化ツールで端末を構築するステップと、DIA記述を用いて端末を記述するステップと、端末にDIA記述を添付するステップと、コンテンツの一部を検索するためにサーバにDIA記述を送り、実行依頼するステップと、サーバがDIA記述を構文解析して処理し、端末によって実行依頼されたDIA記述に一致したフォーマットのコンテンツを選択するステップと、端末にそのコンテンツを伝達するステップとを含む、コンテンツ適合を行う。

【選択図】 図3



【特許請求の範囲】

【請求項 1】

所定の規格に準拠するコンテンツ復号化ツールで端末を構築するステップと、
 D I A（デジタル・アイテム適合）記述を用いて前記端末を記述するステップと、
 前記端末に前記 D I A 記述を添付するステップと、
 コンテンツの一部を検索するために、サーバに前記 D I A 記述を送り、実行依頼するステップであって、前記サーバが前記 D I A 記述を構文解析して処理するステップと、
 前記端末によって実行依頼された前記 D I A 記述に一致したフォーマットのコンテンツを選択するステップと、
 前記端末に前記コンテンツを配信するステップと
 を含む、ユニバーサル・マルチメディア・フレームワーク端末装置のコンテンツ適合方法

10

【請求項 2】

所定の規格に準拠するコンテンツ符号化ツールで端末を構築するステップと、
 D I A（デジタル・アイテム適合）記述を用いて前記端末を記述するステップと、
 前記端末に前記 D I A 記述を添付するステップと、
 前記端末に D I A 記述発生器を実装して、オンラインで D I A 記述を生成可能にするステップと、
 前記端末のネットワーク・プロトコル、他のツール、またはユーザ・プリファレンスに基づいて、ネットワーク条件およびユーザ・プリファレンスを記述した D I A 記述をオンラインで生成するステップと、
 コンテンツの一部を検索するために、サーバに前記 D I A 記述を送り、実行依頼するステップであって、前記サーバが前記 D I A 記述を構文解析して処理するステップと、
 前記端末によって実行依頼された前記 D I A 記述に一致したフォーマットのコンテンツを選択するステップと、
 前記端末に前記コンテンツを配信するステップと
 を含む、ユニバーサル・マルチメディア・フレームワーク端末装置のコンテンツ適合方法

20

【請求項 3】

所定の規格に準拠するコンテンツ符号化ツールで端末を構築するステップと、
 D I A（デジタル・アイテム適合）記述を用いて前記端末を記述するステップと、
 前記端末に前記 D I A 記述を添付するステップと、
 前記端末に D I A 記述発生器を実装して、オンラインで D I A 記述を生成可能にするステップと、
 前記端末のネットワーク・プロトコル、他のツール、またはユーザ・プリファレンスに基づいて、ネットワーク条件およびユーザ・プリファレンスを記述した D I A 記述をオンラインで生成するステップと、
 前記端末の D I D パーサ、I P M P パーサ、R E L パーサ、R D D パーサ、および D I A パーサを実装して、D I D（デジタル・アイテム・デklaration）、I P M P（知的財産権管理および保護）、R E L（権利表現言語）、R D D（権利データ辞書）、および D I A 記述を構文解析するステップと、
 その D I D、I P M P、R E L、R D D、および D I A 記述で D I（デジタル・アイテム）を閲覧するステップと、
 前記端末によって D I D メニューの「チョイス」ノートを選択するステップと、
 前記 D I D メニューに I P M P、R E L、R D D、および D I A 記述が現されている場合には、前記端末の D I D メニューの I P M P、R E L、R D D、および D I A 記述を構文解析するステップと、
 前記 D I D メニューに、適合した D I A ノートが存在するときには、コンテンツを検索するために、前記端末、前記ネットワーク条件、および、前記ユーザ・プリファレンスを記述するのに用いた前記 D I A 記述を、サーバに実行依頼するステップであって、前記サー

30

40

50

バが前記 D I A 記述を構文解析して処理するステップと、
前記端末によって実行依頼された前記 D I A 記述に一致したフォーマットであるコンテンツを選択するステップと、
前記端末に前記コンテンツを配信するステップと
を含む、ユニバーサル・マルチメディア・フレームワーク端末装置のコンテンツ適合方法。

【請求項 4】

トランスコーディング機能を有する適合ゲートウェイを構築するステップと、
D I A パーサを有する適合ゲートウェイを構築するステップと、
端末に添付された D I A (デジタル・アイテム適合) 記述を用いて、前記端末を記述する
ステップと、
前記適合ゲートウェイによってサポートされると想定されている前記端末の各々について、
前記適合ゲートウェイに前記 D I A 記述を登録するステップと、
前記適合ゲートウェイにおいて、その D I A 記述を有するコンテンツを受け取るステップ
と、
前記適合ゲートウェイにおいて、前記 D I A パーサにより、前記コンテンツを記述するの
に用いた、受け取った D I A 記述を構文解析するステップと、
前記コンテンツについて記述するのに用いた、構文解析された前記 D I A 記述を、前記端
末を記述するのに用いた前記 D I A 記述と比較するステップと、
上記ステップにおける前記比較結果により示された、符号化フォーマットに関して一致が
ない場合には、前記コンテンツをトランスコーディングするステップと、
前記ステップによりトランスコーディングされたフォーマットで、前記コンテンツを記述
する、新たな D I A 記述の組を生成するステップと、
前記トランスコーディングされたフォーマットの前記コンテンツに、前記 D I A 記述の組
を添付するステップと、
前記適合ゲートウェイにより、トランスコーディングされた異なる符号化フォーマットを
有するコンテンツを、サポートする前記端末に送信するステップと
を含む、ユニバーサル・マルチメディア・フレームワーク・ゲートウェイ装置のコンテン
ツ適合方法。

【請求項 5】

記憶したマルチメディア・コンテンツでマルチメディア・サーバを構築するステップと、
前記コンテンツの各々に対して D I A 記述を生成するステップと、
前記 D I A 記述を、前記対応するコンテンツに添付するステップと、
前記サーバに D I A パーサを実装するステップと、
前記サーバに適合ツールを実装またはプラグインして、異なるマルチメディアフォーマ
ット間でトランスコーディング機能を実行可能にするステップと、
所定の手段を介して、端末から D I A 記述の組を受け取るステップであって、前記 D I A
記述の組は、前記端末の復号化能力、使用されるネットワークの条件、および、ユーザ・
プリファレンスを記述するのに用いられるステップと、
前記サーバによって前記 D I A 記述の組を構文解析するステップと、
上記ステップにおける構文解析の結果に応じて、前記適合ツールを選択して、トランス
コーディングを行い、前記サーバ内で、あるフォーマットから別のフォーマットに変換する
ステップと、
トランスコーディングされた前記コンテンツを記述する、新たな D I A 記述の組を生成す
るステップと、
メタデータとして、前記新たな D I A 記述の組を、前記トランスコーディングされたコン
テンツに添付するステップと、
を含む、ユニバーサル・マルチメディア・フレームワーク・サーバ装置のコンテンツ適合
方法。

【請求項 6】

記憶したマルチメディア・コンテンツでマルチメディア・サーバを構築するステップと、
前記コンテンツの各々に対してD I A記述を生成するステップと、
前記D I A記述を、前記対応するコンテンツに添付するステップと、
前記サーバにD I Aパーサを実装するステップと、
前記サーバに適合ツールを実装またはプラグインして、異なるマルチメディアフォーマット間でトランスコーディング機能を実行可能にするステップと、
前記サーバにおいて、前記コンテンツを、前記適合ツールを用いて人気がある複数の符号化フォーマットに予めトランスコーディングし、記憶するステップと、
所定の手段を介して、端末からD I A記述の組を受け取るステップであって、前記D I A記述の組は、前記端末の復号化能力、使用されるネットワークの条件、および、ユーザ・プリファレンスを記述するのに用いられるステップと、
前記サーバによって前記D I A記述の組を構文解析するステップと、
前記サーバ内に記憶されたコンテンツであって、前記端末により実行依頼された前記D I A記述を用いて記述された、必要なフォーマットのコンテンツを選択するステップと、
トランスコーディングされた前記コンテンツを記述する、新たなD I A記述の組を生成するステップと、
メタデータとして、前記新たなD I A記述の組を、前記トランスコーディングされたコンテンツに添付するステップと、
一致した符号化フォーマット、ネットワーク条件およびユーザ・プリファレンスを有する前記コンテンツを、前記端末に配信するステップと、
を含む、ユニバーサル・マルチメディア・フレームワーク・サーバ装置のコンテンツ適合方法。

【請求項7】

記憶したマルチメディア・コンテンツでマルチメディア・サーバを構築するステップと、
前記コンテンツの各々に対してD I A記述を生成するステップと、
前記D I A記述を、前記対応するコンテンツに添付するステップと、
前記サーバにD I Aパーサを実装するステップと、
前記サーバに適合ツールを実装またはプラグインして、異なるマルチメディアフォーマット間でトランスコーディング機能を実行可能にするステップと、
視聴およびセレクトのため、ユーザに対して、D I DメニューにI P M P、R E L、R D D、およびD I A記述を提供するステップと、
端末を使用する前記ユーザにより、前記D I Dメニューからのチョイスを行うステップと、
前記チョイス中に前記端末からD I A記述の組を受け取るステップであって、前記D I A記述の組が、前記端末復号化能力を記述するのに用いられるステップと、
前記チョイス中に、前記端末からD I A記述の組を受け取るステップであって、前記D I A記述の組は、前記ネットワーク条件およびユーザ・プリファレンスを記述するのに用いられるステップと、
前記サーバにより、前記D I A記述の組を構文解析するステップと、
上記ステップにおける構文解析の結果に応じて、前記適合ツールを選択して、トランスコーディングを行い、前記サーバ内で、あるフォーマットから別のフォーマットに変換するステップと、
トランスコーディングされた前記コンテンツを記述する、新たなD I A記述の組を生成するステップと、
メタデータとして、前記新たなD I A記述の組を、前記トランスコーディングされたコンテンツに添付するステップと、
一致した符号化フォーマット、ネットワーク条件およびユーザ・プリファレンスを有する前記コンテンツを、前記端末に配信するステップと
を含む、ユニバーサル・マルチメディア・フレームワーク・サーバ装置のコンテンツ適合方法。

10

20

30

40

50

【請求項 8】

記憶したマルチメディア・コンテンツでマルチメディア・サーバを構築するステップと、
 前記コンテンツの各々に対して D I A 記述を生成するステップと、
 前記 D I A 記述を、前記対応するコンテンツに添付するステップと、
 前記サーバに D I A パーサを実装するステップと、
 前記サーバに適合ツールを実装またはプラグインして、異なるマルチメディアフォーマット間でトランスコーディング機能を実行可能にするステップと、
 前記サーバにおいて、前記コンテンツを、前記適合ツールを用いて人気がある複数の符号化フォーマットに予めトランスコーディングし、記憶するステップと、
 視聴およびセレクトのため、ユーザに対して、D I D メニューに I P M P、R E L、R D 10
 D、および D I A 記述を提供するステップと、
 端末を使用する前記ユーザにより、前記 D I D メニューからのチョイスを行うステップと、
 前記チョイス中に前記端末から D I A 記述の組を受け取るステップであって、前記 D I A 記述の組が、前記端末復号化能力を記述するのに用いられるステップと、
 前記チョイス中に、前記端末から D I A 記述の組を受け取るステップであって、前記 D I A 記述の組は、前記ネットワーク条件およびユーザ・プリファレンスを記述するのに用いられるステップと、
 前記サーバにより、前記 D I A 記述の組を構文解析するステップと、
 前記端末によって実行依頼された前記 D I A 記述によって記述される前記要求されたフォーマットである、前記サーバで前記記憶されたコンテンツを選択するステップと、
 トランスコーディングされた前記コンテンツを記述する、新たな D I A 記述の組を生成するステップと、
 メタデータとして、前記新たな D I A 記述の組を、前記トランスコーディングされたコンテンツに添付するステップと、
 一致した符号化フォーマット、ネットワーク条件およびユーザ・プリファレンスを有する前記コンテンツを、前記端末に配信するステップと
 を含む、ユニバーサル・マルチメディア・フレームワーク・サーバ装置のコンテンツ適合方法。

【請求項 9】

請求項 1 に記載のユニバーサル・マルチメディア・フレームワーク端末装置のコンテンツ適合方法であって、
 D I A 記述を用いて前記端末を記述するステップが、
 端末の一般的特徴、端末ハードウェア、端末ソフトウェア、端末システム、および、第 1
 のレイヤとして、端末復号化能力を含む、階層構造を用いて前記端末を記述するステップ
 と、
 主要コンテンツ、ソース復号化レイヤ、システム復号化レイヤ、伝送レイヤと異なる符号
 化レイヤにおいて、前記 D I A 記述で前記端末復号化能力を記述するステップと、
 符号化および復号化パラメータを含む符号化レイヤの各々において、詳細な記述を規定し
 て、前記復号化処理で用いられる異なるパラメータに対して、前記端末によりサポートさ
 れる最大範囲を示すステップと、
 上述の D I A 記述で用いる制御語であって、共通語の組であり、前記端末と前記サーバと
 の間で信号を出すのに用いられる制御語を規定するステップと
 を含む、ユニバーサル・マルチメディア・フレームワーク端末装置のコンテンツ適合方法
 。

【請求項 10】

請求項 1 に記載のユニバーサル・マルチメディア・フレームワーク端末装置のコンテンツ適合方法であって、
 D I A 記述を用いて前記端末を記述するステップが、
 端末の一般的特徴、端末ハードウェア、端末ソフトウェア、端末システム、および第 1 の
 レイヤとして端末復号化能力を含む、階層構造を用いて前記端末を記述するステップと、
 50

主要コンテンツ、ソース復号化レイヤ、システム復号化レイヤ、伝送レイヤと異なる符号化レイヤにおいて、前記D I A記述で前記端末復号化能力を記述するステップと、符号化および復号化パラメータを含む符号化レイヤの各々において、詳細な記述を規定して、前記復号化処理で用いられる異なるパラメータに対して、前記端末によりサポートされる最大範囲を示すステップと、

上述のD I A記述で用いる制御語であって、共通語の組であり、前記端末と前記サーバとの間で信号を出すのに用いられる制御語を規定するステップと

拡張性のために、XMLで前記D I A記述を表現するステップと

を含む、ユニバーサル・マルチメディア・フレームワーク端末装置のコンテンツ適合方法。

10

【請求項 1 1】

請求項 1 に記載のユニバーサル・マルチメディア・フレームワーク端末装置のコンテンツ適合方法であって、

D I A 記述を用いて前記端末を記述するステップが、

端末の一般的特徴、端末ハードウェア、端末ソフトウェア、端末システム、および、第 1 のレイヤとして、端末復号化能力を含む、階層構造を用いて前記端末を記述するステップと、

主要コンテンツ、ソース復号化レイヤ、システム復号化レイヤ、伝送レイヤと異なる符号化レイヤにおいて、前記D I A記述で前記端末復号化能力を記述するステップと、

符号化および復号化パラメータを含む符号化レイヤの各々において、詳細な記述を規定して、前記復号化処理で用いられる異なるパラメータに対して、前記端末によりサポートされる最大範囲を示すステップと、

20

上述のD I A記述で用いる制御語であって、共通語の組であり、前記端末と前記サーバとの間で信号を出すのに用いられる制御語を規定するステップと

S D L (構文記述言語) の形式で前記D I A記述を生成するステップと

を含む、ユニバーサル・マルチメディア・フレームワーク端末装置のコンテンツ適合方法。

【請求項 1 2】

請求項 2 に記載のユニバーサル・マルチメディア・フレームワーク端末装置のコンテンツ適合方法であって、前記端末内にD I A記述生成部を実装して、オンラインでD I A記述を生成するステップを有し、

30

帯域幅、遅延およびパケット損失の制御語を規定することにより、コンテンツフォーマット適合に関連するネットワーク条件を記述し、

品質、色、大きさ、ストリーミング、ダウンローディング、それらの属性によるフィルタリングの制御語を規定することにより、コンテンツ・フォーマット適合に関するユーザ・プリファレンスを記述し、さらに、

ネットワーク・プロトコル、他のプライベート・ツール、またはユーザ入力から、前記端末においてネットワーク条件を受け取るステップと、

上述のステップにおいて規定された前記制御語を用いることにより、前記端末において、ネットワーク条件に関する、受け取ったパラメータを、前記D I A記述に変換するステップと、

40

前記端末において、ユーザ・インタフェースからのユーザ・プリファレンスを受け取るステップと、

上述のステップにおいて規定された前記制御語を用いることにより、前記端末において、ユーザ・プリファレンスに関する、受け取ったパラメータを、前記D I A記述に変換するステップと

を含む、ユニバーサル・マルチメディア・フレームワーク端末装置のコンテンツ適合方法。

【請求項 1 3】

ネットワークに接続されたユニバーサル・マルチメディア・フレームワーク端末装置であ

50

って、
コンテンツ複合化ツールが記録されているメモリ装置と、
D I A 記述を用いて前記端末を記述する D I A 記述生成器と、
D I A 記述を前記端末に添付する添付部と、
一つのコンテンツを検索するために前記 D I A 記述とコンテンツ要求とをサーバに配信する伝送器と、
前記サーバで選択され、前記サーバから配信された前記コンテンツを受信する受信機とを備えたユニバーサル・マルチメディア・フレームワーク端末装置。

【請求項 14】

端末とサーバとの間に設けられたユニバーサル・マルチメディア・フレームワーク・ゲートウェイであって、

メモリ装置と、

D I A 記述を用いて前記端末を記述する D I A 記述生成器と、

D I A 記述を前記端末に添付する第 1 添付部と、

前記端末に添付されている D I A 記述を前記メモリ装置に登録するレジスタと、

前記端末によるコンテンツ要求に対応するコンテンツを前記サーバから受信する受信機と

、
前記コンテンツに添付され、受信した D I A 記述を構文解析する D I A パーサと、
構文解析された D I A 記述を前記端末についての前記 D I A 記述と比較する比較器と、
前記コンテンツのフォーマットと前記端末の D I A 記述に記載されたフォーマットとのミスマッチがある場合には、前記コンテンツのフォーマットを前記端末の D I A 記述に記載されたフォーマットにトランスコードするトランスコーダと、
前記トランスコードされたコンテンツを記述する新たな D I A 記述の組を生成する生成器と、

前記新たな D I A 記述の組を前記トランスコードされたコンテンツに第 2 添付部と、

前記コンテンツを前記端末に配信する伝送器と

を備えるユニバーサル・マルチメディア・フレームワーク・ゲートウェイ。

【請求項 15】

ネットワークを介して端末に接続されているユニバーサル・マルチメディア・フレームワーク・サーバであって、

マルチメディアコンテンツが記録されているメモリ装置と、

各コンテンツについての D I A 記述を生成する D I A 生成器と

対応するコンテンツに前記 D I A 記述を添付する第 1 添付部と、

端末の復号化能力、ネットワーク条件、及びユーザ・プリファレンスについて記述するために用いられる D I A 記述の組を前記端末から受信する受信機と、

前記受信した前記端末に関する D I A 記述の組を構文解析する D I A パーサと、

前記コンテンツを異なるマルチメディア・フォーマット間でトランスコードする適合ツールと、

前記構文解析した結果に従ってトランスコードして一つのフォーマットから他のフォーマットに変換する一つの適合ツールを選択するセクタと、

前記トランスコードされたコンテンツを記述する新たな D I A 記述の組を生成する生成器と、

前記新たな D I A 記述の組を前記トランスコードされたコンテンツにそのメタデータとして添付する第 2 添付部と、

前記コンテンツを前記端末に配信する配信部と

を備えるユニバーサル・マルチメディア・フレームワーク・サーバ。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、異なるフォーマットのマルチメディア・コンテンツをアクセスできる統一マル

10

20

30

40

50

チメディア端末に関し、かつ異なるマルチメディア端末、ネットワーク、使用環境を取扱うことができる統一マルチメディア・フレームワークに関する。

【0002】

【従来の技術】

MPEGおよび他の規格体は、効率的な方法である場所から別の場所へのコンテンツ転送および配信、並びに限定された空間における大きな容量を伴うコンテンツ記憶を容易にするために、ビデオ、オーディオ、システム、通信プロトコル、コンテンツ提示、コンテンツ・パッケージング、等における多くの規格を生成した。

【0003】

その結果として、既存のコンテンツは、異なるソース符号化方法、MPEG1、MPEG2、MPEG4、DV、DVCPRO等による、異なるフォーマットで見受けられ、同時にデバイスおよび端末は、また、ある一定の標準に準拠するが別の標準には準拠しない異なる範囲で構築される。例えば、DTV STB（セットトップボックス）は、MPEG-2に準拠したコンテンツを復号化しかつ再生できるが、DVコンテンツ、またはMPEG-4コンテンツを再生することができない。

【0004】

どのようなフォーマットであってもコンテンツをアクセスし、かつ使用できる、普遍的なマルチメディア端末が存在することが望ましい。これを達成するための少数の方法が存在する。

【0005】

一つの方法は、全ての異なるフォーマットを理解しかつリアルタイムでそれらを復号化し、そしてリアルタイムで要求されたフォーマットにコンテンツをコード化するための強力な変換符号化器（トランスコーダ）の機能を有する端末を構築することである。しかし、そのようなリアルタイム多機能トランスコーダ機能を有すると、非常にコストが高くなる。DTV、またはDVDのようなデバイスに対しても、ポータブル端末に対して使用することは、まったく望ましくない。

【0006】

別の方法は、異なるフォーマットで同じコンテンツを予めトランスコードしかつ記憶するために、サービス・プロバイダ側／コンテンツ生成側、またはゲートウェイ・サイトにそのようなトランスコーダ機能を有することである。

【0007】

そのような場合には、コンテンツ提示フォーマットを記述する記述またはメタデータの組を生成し、かつ対応するコンテンツに添付できる。端末によって依頼された問合せを、サーバに記憶されたメタデータと一致させることによって、サーバは特定フォーマットのどのコンテンツが端末に配信されるのかを決定する。ここで、端末は、サーバが認識できる定義された記述またはメタデータの組を発生できるように、準拠した端末でなければならない。

【0008】

コンテンツ提示フォーマットをアドレス指定するための記述の組の他に、端末を記述するための記述の組、並びにネットワーク条件およびユーザ・プリファレンスを含んでいる他の使用環境を記述するための一組の記述を定義することもまた必要である。しかしながら、使用環境は、コンテンツ提示フォーマットおよび端末により異なり、かつそれは、変更可能かつ選択可能であり端末で符号化性能のように端末で固定されない。

【0009】

ここで、本発明は、記述および記述中の構造の組を定義すること、並びに異なる端末およびネットワーク間で用いられるマルチメディア・フレームワークに対する機構を定義することに焦点をおく。

【0010】

【発明が解決しようとする課題】

本発明は、以下の問題の解決を試みる。

10

20

30

40

50

【0011】

ある一定のフォーマットの同じコンテンツまたはメディア・リソースは、異なるネットワーク条件および異なるユーザ・プリファレンスを有するマルチメディア端末によってアクセスされ、検索し、かつ使用できる。

【0012】

異なる復号化性能およびネットワーク条件を有する端末は、異なるデータ・フォーマットであるコンテンツまたはメディア・リソースをアクセスし、検索し、かつ消費できる。

【0013】

【課題を解決するための手段】

コンテンツ提示フォーマットに対する記述を規定することによって、コンテンツまたはメディア・リソースは、その符号化フォーマットおよびスタイルを反映すべく記述できる。このとき、この記述は、所定の符号化規格による符号化条件に基づき、そしてまたビット・ストリーム・ヘッダから抽出したパラメータに基づき定義される。

【0014】

端末に対する記述の組を定義することによって、端末は、準拠した復号化性能を反映するように記述できる。

【0015】

端末に対する記述の組を定義することによって、端末は、その処理パワー、メモリ、オペレーティング・システム等を反映するように記述できる。そのような記述は、サーバが正しい符号化フォーマットでコンテンツを配信する前に端末からサーバに送られる。

【0016】

ネットワークに対する記述の組を定義することによって、端末が接続されるネットワーク条件は、通信プロトコル、平均帯域幅、遅延状態、パケット損失状態等を反映するように記述できる。

【0017】

ユーザ・プリファレンスに対する記述の組を定義することによって、ユーザの関心および好み（プリファレンス）は、ユーザのチョイスおよびセレクションを反映すべく記述できる。

【0018】

上記記述の階層および構造を定義することによって、マルチメディア・フレームワークは、ネットワークを介して端末へコンテンツを配信するために構築できる。

【0019】

上記記述の階層および構造を定義することによって、端末は、限定された復号化性能を有するユニバーサル・マルチメディア端末を形成するように構築できる。

【0020】

マルチメディア・フレームワークを構築するために用いられた機構（仕組み）を定義することによって、既存および将来のアプリケーションの多くは、異なる端末および異なるネットワークを有するより多くのユーザに供される。

【0021】

【作用】

端末側では、

1) MPEG-21 端末は、シンプル・プロファイルに対する MPEG-4 ビデオ・デコーダのようなソース符号化ツールの一つを実装することによって構築される。

【0022】

2) MPEG-21 端末は、MP4 ファイル・フォーマットのようなシステム符号化ツールの一つを実装することによって構築される。

【0023】

3) MPEG-21 端末は、RTP のような伝送ツールの一つを実装することによって構築される。

【0024】

10

20

30

40

50

4) M P E G - 2 1 端末は、M P E G - 2 1 D I D (デジタル・アイテム・デクラレーション)、R E L (権利表現言語)、R D D (権利データ記述)、I P M P (知的財産権管理および保護)、およびD I A (デジタル・アイテム適合) パーサを実装することによって構築される。

【0025】

5) M P E G - 2 1 端末は、より詳細な記述がビット・レート、ディスプレイ・ウィンドウ・サイズ、バッファ・サイズ、等のような、各タイプの符号化フォーマットに含まれるような、テーブルがX M L スキーマまたはS D L (構文記述言語) 構文で表現できるような、ソース符号化フォーマット、システム・レイヤ符号化フォーマット、伝送フォーマットを含んでいる準拠した復号化フォーマットのような端末の特性を記述すべくテーブルをセットアップすることによって構築される。

【0026】

コンテンツ・サーバ側：

1) ある一定のデータ・フォーマットであるコンテンツまたはメディア・リソースは、M P E G - 2 1 D I A 定義された構造により各タイプのフォーマットを記述すべくその定義されたメタデータまたは一組の定義されたD I A 記述で異なるデータ・フォーマットに予めトランス・コーディングされる。

【0027】

2) M P E G - 2 1 デジタル・アイテム (D I) は、シーン構造を構築することにより生成される。ここでは、対応するD I D / I P M P / R E L / R D D / D I A 記述が、構造的に一緒にリンクされ、デジタル・アイテムの内容、およびメディア・リソースの所在、保護形態、D I を用いる権利の内容、および、利用可能な端末およびネットワーク適合を示す。この適合は、メディアリソースを「チョイス」する形で列挙され、ユーザにより配信され、使用される。これは、双方向インタラクティブ・アプリケーションに用いられる。

【0028】

3) 放送のような一方向アプリケーションに対して、M P E G - 2 1 デジタル・アイテムは、端末から受け取った特定の端末およびネットワーク条件により生成される。

【0029】

なお、用語の定義は以下に示す通りである。

デジタル・アイテム：デジタル・アイテムは、I S O / I E C 21000 フレームワークにおける標準表現、識別表示、及びメタデータを備える。また、このエンティティは、このフレームワークの流通及び取引の基本的なユニットである。

デジタル・アイテム適合：デジタル・アイテムがリソース適合エンジン及び／又はディスクリプタ・エンジンの対象とされる処理であって、それによって適合デジタル・アイテムを生成する。

【0030】

コンテンツ・デジタル・アイテム：リソースの配信に用いられるデジタル・アイテムである。

【0031】

【発明の実施の形態】

従来技術が、図1および図2に示されており、あるデータ・フォーマットのコンテンツが異なる端末によって再生できない現在の状況を説明する。

【0032】

図1では、一方向でM P E G - 2 M P @ M L コンテンツがモジュール1. 1のマルチメディア・サーバからユーザに配信され、かつモジュール1. 4のD T V S T B だけが再生できるということが示されている。モジュール1. 2、1. 3、および1. 5の他の端末は、コンテンツを演奏（上映）できない。

【0033】

図2では、インタラクティブ双方向でM P E G - 2 M P @ M L コンテンツがモジュール2

10

20

30

40

50

、1のマルチメディア・サーバからユーザに配信され、かつモジュール2.4のDTVSTBだけが再生できることが示されている。モジュール2.2、2.3、および2.5の他の端末は、コンテンツを演奏（上映）できない。

【0034】

上記例から、そのような状況は、より広い範囲の端末およびネットワークによるコンテンツ・アクセス性を制限するということが理解される。

【0035】

他方、端末は、非常にコストがかかるため、コンテンツの全ての異なるフォーマットを復号化可能なようには構築できない。

【0036】

以下の実施形態は、二つの主要部分で構成される。すなわち、インタラクティブ双方向または一方向アプリケーションで用いられる、異なる端末への適合と、異なるネットワークへのリアルタイム・ストリーミング適合に対する、階層DIA記述(Descriptions)という二つの主要部分で構成される。

【0037】

インタラクティブ双方向アプリケーションの場合

DTVインタラクティブ・データ放送、インターネット・ベース・サービス、インタラクティブDVD、携帯電話によるインタラクティブ・データ・アクセス等、インタラクティブ・アプリケーションは、既存の世界では非常に人気がある。

【0038】

この双方向アプリケーションに対する一般的なモデルは、サーバがユーザによって視聴されるレイヤ構造化メニューを提供し、ユーザがクライアント端末からサーバに要求を送ることによってユーザが検索したい事項を選択し、かつサーバが要求を処理して、クライアント端末に要求されたコンテンツを配信することである。

【0039】

ここで、コンテンツは、MPEG-21でメディア・リソースと呼ばれる。

【0040】

図3に示すように、MPEG-21 マルチメディア・サーバは、モジュール3.1に示されている。一組のDIA記述が添付された、コンテンツまたはメディア・リソースが記憶される。

【0041】

モジュール3.11では、MPEG-21端末が示されている。MPEG-21端末は、モジュール3.10のソース符号化ツール、モジュール3.9のシステム・レイヤ符号化ツール、モジュール3.8の伝送プロトコル、モジュール3.7のMPEG-21 DIAパーサ、および、DIA記述の組で構成される。端末は、モジュール3.6のTCP/IP、UDP、ATM、または他のネットワークに接続される。

【0042】

ここで説明しているケースは、ケース2の、図3に示す双方向インタラクティブ・アプリケーションに対応する。最初に、端末またはユーザは、デジタル・アイテムを閲覧し、モジュール3.5でユーザが選択する事項に基づきDID/IPMP/REL/RDDおよびDIA記述を構文解析する。サーバは、モジュール3.4で、端末によってなされた選択により、一致したデータ・フォーマットのコンテンツを端末に配信する。

【0043】

上記インタラクティブ処理では、サーバによって受け取られた情報は、二つのタイプ、すなわち、ユーザ関連情報、およびユーザ非関連情報で構成される。

【0044】

1) ユーザ関連情報：ユーザ・プリファレンスのような、閲覧中にユーザによってなされた要求または選択

【0045】

2) ユーザ非関連情報：DIA記述が端末のDIAパーサを通して構文解析された際に自

10

20

30

40

50

動的に端末から送信された端末特性。

【0046】

一方向放送アプリケーションの場合

この方法では、端末は、双方向アプリケーションのようにインタラクティブにサーバと通信することが想定されていない。

【0047】

ケース1の一方向アプリケーションについて図3に示されているように、モジュール3.2で、端末は、DIA記述の組を有する要求を送信することにより、コンテンツ配信を初期化し、身元と能力をサーバに知らせる必要がある。サーバは、その要求およびDIA記述の組を所有し、モジュール3.3で、一致したフォーマットのコンテンツを伝送する。

10

【0048】

DIA記述には二つのタイプが存在する。一方のタイプは、端末準拠復号化フォーマット、端末処理パワー等の、端末特性に関する。このタイプの記述は、端末で固定されかつユーザによって選択されない。他方のタイプは、端末で固定されていないネットワーク条件およびユーザ・プリファレンスに関する。これらは、異なるネットワーク条件および異なるユーザ・プリファレンスに変更可能である。

【0049】

一方向または双方向アプリケーションのいずれでも、端末の特性に関するこれらのDIA記述は、ユーザによってではなく、端末により、サーバに送信されなければならない。一方向アプリケーションに対してDIA記述は、通信開始期間中にサーバに実行依頼することができ、双方向アプリケーションに対してDIA記述は、インタラクティブ通信の開始または中間地点でサーバに実行依頼できる。ネットワーク条件およびユーザ・プリファレンスに関するそれらのDIA記述は、伝達処理中に通信で用いられるネットワーク・プロトコルまたはユーザによって与えることができる。

20

【0050】

各符号化レイヤに対する階層記述

図4に示すように、モジュール4.1の汎用メディア・リソースは、モジュール4.2でソース符号化フォーマットとして考慮される、テキスト、グラフィックス、静止画像、オーディオ、ビデオの形式でありうる。

【0051】

オーディオおよびビデオが組み合わされている場合には、モジュール4.3に示すようなシステム符号化が要求され、製品において広く用いられているMPEG-2トランスポート・システムおよびプログラム・システムのような、オーディオおよびビデオを同期させる。

30

【0052】

リアルタイム伝送のような、ある伝送目的を実現する、モジュール4.4に示す伝送プロトコルが要求されることがある。

【0053】

そのようなメディア・リソースは、様々なタイプの符号化フォーマットを用いて形成され、かつ記述の組は、MPEG-21DIA記述と呼ばれる、モジュール4.5に示すように、メディア・リソースの符号化構造を告げるために生成される。

40

【0054】

そのような記述および異なるレイヤ中のレイヤ構造は、XMLまたはSDL（構文記述言語）の形式のMPEG-21ファイル・フォーマットで表すことができる。ここでは、考えられる解の一つとしてそのようなDIA記述を記述するためにXMLベース・スキーマを用いる（スキーマの構造を図5および図6に示しかつ詳細なスキーマをこのセクションの終りに見出すことができる）。

【0055】

図5および図6のスキーマ構造では、我々は、「汎用」記述（5.1）、「ハードウェア」（5.2）、「ソフトウェア」（5.3）、および「システム」（5.4）特性を含ん

50

でいる余すところのない端末記述を定義する。端末符号化能力の階層記述は、「符号化能力」(Coding Capability) (5.5)で見ることができる。

【0056】

汎用端末記述では、端末ベンダー、モデル、およびエンコーダ、デコーダ、ゲートウェイ、ルータ、PC、PDA、プリンタ、携帯電話等のような、そのタイプ/クラスが記述される。

【0057】

ハードウェアおよびソフトウェア・コンポーネントは、端末能力を示すための二つの重要な要素である。これら二つの部分に関する各要素の簡単な記述を以下に掲げる。

【0058】

CPU:ベンダー、モデル、デバイスCPUのスピード、例えば、インテル ペンティアム (登録商標) 111 1.13GHz;

【0059】

メモリ:ベンダー、モデル、デバイス・メモリのサイズ、例えば、インテル SDRAM PC133 128M;

【0060】

スピーカ:ベンダー、デバイス・スピーカのタイプ、例えば、ヤマハ YMF753;

【0061】

キーボード:ベンダー、デバイス・キーボードのタイプ、例えば、東芝 85-キー エルゴノミック・キーボード;

【0062】

画面:端末画面のより詳細を与える。「サイズ」:水平および垂直値で構成された、画素単位でのデバイス画面サイズ;「サイズ文字」:水平および垂直値で構成されかつデバイスの標準フォントによって決定された、文字単位でのデバイス画面サイズ;「画素毎のビット」:デバイスが画面に表示できるグレーのカラーまたはシェードの数に関する、画素毎のカラーまたはグレースケール情報のビット数;「位置決め解像度」デバイス画面によってサポートされる位置決め付属機構の解像度のタイプ。

【0063】

ハードウェア・サポート:端末が「カラー」(カラー表示)、「テキスト入力」(テキスト入力)、「画像入力」(画像表示)、「音声入力」(あらゆる形式の音声入力)、「ソフトキー」(プログラム・ソフト・キー)および「サウンド出力」(外部スピーカを介するサウンド出力)をサポートするかどうかを示す。全てのこれらのサポートを示すために、ブール・タイプが用いられる、例えば、「カラー」サポートに対する「はい」は、デバイスの表示がカラーをサポートすることを意味する。

【0064】

アシスト・ハードウェア:セキュリティ機能性をサポートしているアシスタント・ハードウェア、「スマートカード」、「ハードキー」、

【0065】

OS:ベンダー、名前、バージョン、端末のオペレーティング・システムの通し番号、例えば、アップル マック OS、9.0、12345678、

【0066】

VM:ベンダー、名前、デバイスに組込まれた仮想マシンのバージョン、例えば、Sun JRE1.2、

【0067】

ファームウェア:ベンダー、名前、デバイスの低レベル・ソフトウェアが準拠するファームウェアのバージョン、

【0068】

ソフトウェアサポート:端末が「その名前およびバージョンを有する」、「ソフトウェア・ダウンロード」、「ブラウザ」をサポートするかどうかを示す。

【0069】

10

20

30

40

50

また、システム特性も端末記述に対して重要な形態である。

【0070】

端末システムの汎用記述は、「モジュール」を処理すること、例えば、ある特殊機能性の実行に対するエクストラ端末ツール、「コンポーネント」の相互接続および「構成」オプションのサポートを含んでいる。MPEG-21 マルチメディア・フレームワークに対して、特殊システム特性「IPMP」が供給されるべきである。IPMPシステム記述は、それらのツール（Tool）IDおよびRDD/RELのサポートを有する常駐IPMP_Toolsを含む。

【0071】

端末符号化性能は、コンテンツ・タイプ（オーディオ、ビデオ、画像／グラフィック、テキスト）および各符号化タイプ（例えば、MPEG、ITU）による構造化モードで記述される。各符号化タイプでは、更に分類しかつ拡張するためにプロファイルおよびレベルが用いられる。これらの符号化フォーマット記述の他に、画像サイズ、表示ウィンドウ・サイズ、ビット・レート、フレーム・レート、バッファ・サイズ、等のような符号化パラメータ（「コントローリング・ターム」）は、符号化フォーマットを記述するために用いる可能な制御アイテムである。詳細は、以下のビデオ符号化システム構造化説明および端末スキーマ記述で見出すことができる。

【0072】

また、ビデオを記述するために要求される可能なアイテムまたはパラメータが何であることを説明するためにビデオに対する詳細な構造化を図4に示す。

【0073】

MPEG-nビデオ符号化フォーマットに対するように、MPEG1ビデオ、MPEG2ビデオ、MPEG4ビデオ、そしてまたMPEG4 AVC（アドバンス・ビデオ符号化）が存在する。各MPEGビデオ符号化フォーマットに関してプロファイル、レベル、準拠レベルをを告げるためのバージョンが存在する。

【0074】

所与のメディア・リソースに対して、そのビデオ符号化フォーマットをちょうど記述するためにプロファイルが何か、レベルが何か、およびバージョンが何かを与えなければならない。図4のモジュール4.6、4.7および4.8においてそれぞれ示されているように、レベル1およびバージョン1を有するシンプル（Simple）・プロファイルがビデオ符号化に対して示されている。

【0075】

そのトップには、また、画像サイズ：W／H（幅および高さ）、表示ウィンドウ・サイズ：W／H、ビット・レート、フレーム・レート、バッファ・サイズ、および、もしあればその他の事項がビデオ符号化フォーマットを記述するために用いるべく可能なアイテムであるような、モジュール4.9に示される、DIAビデオ符号化記述に含まれるべく符号化パラメータが要求される。

【0076】

ここで、ビデオは、CCIR601 4：2：2、各カラー・コンポーネントに対して8ビットを有するRGB、等のような、異なるビデオ・フォーマットの主要ビデオを含むことができる。

【0077】

また、ビデオ・フォーマットは、DVカメラで用いられるDV（デジタル・ビデオ（Digital Video））フォーマット、DVCPRO-25、DVCPRO-50、DVPRO-HDを含むDVxxxもカバーする。これらは、オリジナル・コンテンツで見出されかつデジタル・カメラによってキャプチャされる人気があるフォーマットである。

【0078】

H.26xは、H.261およびH.263で構成され、かつテレビ会議製品で主に用いられる。

10

20

30

40

50

よびトランスコーディング・モジュールで構成される。

【0090】

図8に示すように、モジュール7.1では、適合ゲートウェイのDIAパーサは、サーバから受け取った入力DIA記述を処理し、モジュール7.1の後では、処理した結果は、モジュール7.2で端末によって作成された端末のDIA記述と比較される。モジュール7.2における結果に基づき、ソース符号化レイヤおよびシステム符号化レイヤは、オーディオ、ビデオ、およびシステム符号化フォーマットに対してそれぞれモジュール7.3、7.4および7.5で実行される。

【0091】

しばしば計算時間を節約するために、変換またはトランスコーディングは、ビット・ストリーム・レベルで行うことができる。そのようなインテリジェント・トランスコーディング・ツールは、端末でも実装することができる。実装は、状況に応じて異なりうる。これらは、実施する者に託され、それらの端末の処理能力およびアプリケーションに応じて考慮する。

【0092】

目的にかなう別のより良い方法が存在する。有用なワーキング・モデルは、サーバ側の処理能力に依存すべきである。より重要な点は、メディア・リソースを予めコード化しかつマルチメディア・サーバまたはライブラリに記憶するアプリケーションの方針である。そのような場合には、同じコンテンツを予めトランスコードして異なる符号化フォーマットに記憶でき、人気がありかつ広く用いられる場合に適合できる。

【0093】

実施形態の一つとして、一般的なブロック図を図9に示す。そのDIA記述を有するMP EG-21端末モジュール8.1は、図7のモジュール6.2に示す端末準拠符号化フォーマットを記述するDIA記述の組とともに、コンテンツ要求を、サーバに送っている。サーバは、モジュール8.2で受け取ったDIA記述を処理し、かつモジュール8.3で要求されたコンテンツに対して一致した符号化フォーマットを探索する。そして、要求に適合した符号化フォーマットのコンテンツが、サーバから端末に配信されている。これは、放送に似た一方向アプリケーションの場合である。

【0094】

インタラクティブ・アプリケーションについての別の実施形態を、図10に示す。モジュール9.1の端末は、(1)においてDID/REL/RDD/IPMP/DIAで記述されるデジタル・アイテムを閲覧する。サーバは、異なる「チョイス」を与えて、(2)において端末特性、ネットワーク条件、またはユーザ・プリファレンスを示す。

【0095】

サーバが端末特性を獲得する二つの方法が存在する。ユーザが、一旦、図10に示す(3)において、ユーザに表示された「端末特性」を選択した場合には、端末は、サーバにそのDIA記述を自動的に送信する。別の方法は、ユーザがいずれかの手段によってそれらの端末について知っている場合に端末特性をユーザに決定させる。そして、適合したメディア・リソースとして、一致した符号化フォーマットを有するコンテンツは、(4)において端末に配信される。

【0096】

端末特性は、準拠した符号化能力、および、その処理パワー/メモリ/オペレーティング・システム等をも含む。端末の準拠した符号化能力は、メディア・リソースを記述するのに用いられるコンテンツ提示フォーマットに対応している。それらは、同じ記述であるが、異なるターゲット・オブジェクト(端末またはコンテンツ/メディア・リソース)に対して異なる語を用いている。

【0097】

異なるネットワークへのリアル・タイム・ストリーミング適合

上述したように、二つの異なるタイプのDIA記述が存在する。すなわち、固定されかつ変更できない端末特性を記述する固定記述、および、端末により固定されない、ネットワ

10

20

30

40

50

ーク条件およびユーザ・プリファレンスを記述する変更可能記述 (Changeable Descriptions) である。D I A 記述は、使用されるネットワークまたはユーザ・プリファレンスが異なるか変更されれば異なる。

【0098】

適合を指示する階層構造および機構を有する前者の記述は、上のセクションにおいて説明されている。後者の記述およびサーバとクライアント端末との間に信号を送る機構は、このセクションに記載されている。

【0099】

同じ符号化フォーマットがメディア・リソースおよび端末に対して用いられると仮定する。帯域幅が異なるといった、異なるネットワーク条件により、何らかの手段により、変更可能な条件に適合する必要がある。この手段は、ソース符号化レイヤおよびまたアプリケーション・ネットワーク・レイヤにおいてスケーラブル符号化技法の使用を含む。

【0100】

変更可能なネットワークに適合すべくリアルタイムでビット・レートを変更する多くの方法が存在する。これらは、S N R スケーラブル符号化、ビット・レート・コントローラ調整、フレーム・レートの低減、エンハンスメント・レイヤに対するビット切り捨て、Q o f (サービス品質) の利用等を含む。これらの全ては、異なる種類の適合ツールとして考えられ、かつそれらは、場合に応じて異なる。

【0101】

図11では、リアルタイム適合フレームワークを示す。これは、ストリーミングの場合を示しており、ネットワーク条件およびユーザ・プリファレンスが常にサーバにフィードバックされる。それにより、サーバはそのような情報を使用して、どのような種類のメディアリソースが端末に配信されて、適切なビットレートに適合するかを決定できる。

【0102】

図11に示すように、D I A パーサ、適合ツール、およびメディア・リソースが含まれるモジュール10. 1に、M P E G - 2 1 マルチメディア・サーバが示される。M P E G - 2 1 端末は、モジュール10. 5のD I A 記述発生器を伴うモジュール10. 2で示されており、モジュール10. 3のネットワーク条件およびユーザ10. 4のユーザ・プリファレンスに応じて、D I A 記述を生成する。

【0103】

モジュール10. 6では、記述の組は、サーバに実行依頼するのに必要な、考えられるアイテムが何であるかを示す例として列挙されている。

【0104】

サーバのモジュール10. 7のD I A パーサは、クライアント端末から受け取ったD I A 記述を処理し、モジュール10. 8の適合ツール・プラグインからの適切な適当な適合ツールの選択を決定する。モジュール10. 9のメディア・リソースは、適合ツールによって変換され、D I A 記述によって記述される要求された適当なネットワーク条件およびユーザ・プリファレンスに適合する。そして、要求されたメディア・リソースは、サーバからクライアント端末に受け渡される。

【0105】

一般的な適合フレームワークを図12に示す。図12は、限定された復号化能力、異なるネットワーク条件およびユーザ・プリファレンスを有する所与の端末に対して、どのように適合が実行されるかを示す。

【0106】

M P E G - 2 1 マルチメディア・サーバを、D I A パーサを有するモジュール11. 1に示す。異なる適合ツールは、サーバ側でプラグイン・ツールとして実装され、または用いられる。コンテンツは、要求に基づき検索されかつユーザに受け渡されるべきメディア・リソースとしてサーバに記憶される。

【0107】

モジュール11. 6にその復号化能力および処理能力を記述するためのD I A 記述、並び

10

20

30

40

50

にモジュール 11. 5 にネットワーク条件およびユーザ・プリファレンスの D I A 記述を発生する D I A 記述オンライン発生モジュールを含む、モジュール 11. 2 に M P E G - 2 1 クライアント端末を示す。

【0108】

通信の起動の間、端末は、コンテンツ配信を開始するために、「a」の矢印で示すように、サーバに端末性能の一組の D I A 記述を送る。

【0109】

モジュール 11. 3 のネットワーク条件およびモジュール 11. 4 のユーザ・プリファレンスは、それが要求された場合にはいつでも D I A 記述オンラインを発生するためにモジュール 11. 5 に実際のネットワーク条件およびユーザ・プリファレンスをフィードバックする。端末は、「b」の矢印で示すように、サーバにネットワーク条件およびユーザ・プリファレンスの一組の D I A 記述を送る。

【0110】

ユーザ・プリファレンスを得るためにモジュール 11. 10 にユーザ・インタフェースを示す。ネットワーク条件は、ネットワーク・プロトコル、またはモジュール 11. 11 の適当な方法から得ることができ、かつユーザ・インタフェースを介してユーザによって与えられることもまた可能である。

【0111】

モジュール 11. 7 に示すように、サーバ側の D I A パーサは、端末から受け取った D I A 記述を処理し、かつ端末性能および適当なネットワーキング条件およびユーザ・プリファレンスに一致すべく要求されたメディア・リソースを一致したメディア・リソースに変換しかつトランスコードすべくサーバからある一定の適合ツールまたはプラグイン・ツールを選択する。

【0112】

適合ツールは、サーバ側のモジュール 11. 8 に示され、かつそれは、また、図 8 に示すように中間適合ゲートウェイに実装することもできる。モジュール 11. 9 に示したメディア・リソースは、サーバに記憶され、かつそれは、同時に多くのユーザに対するリアルタイム・トランスコーディングを回避するために、複数の人気がある符号化フォーマットをターゲットすべく予めトランスコードされ、サーバに記憶できる。

【0113】

以下に、詳細な端末記述スキーマを示す。

10

20

30

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v4.3 U (http://www.xmlspy.com) by Huang Zhongya
ng (Panasonic Singapore Laboratories Pte Ltd) -->
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault
="qualified" attributeFormDefault="unqualified">
  <xs:element name="TERMINAL">
    <xs:annotation>
      <xs:documentation>It give the description of terminal capabilities</x
s:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:sequence maxOccurs="unbounded">
        <xs:element name="General" minOccurs="0">
          <xs:complexType>
            <xs:sequence maxOccurs="unbounded">
              <xs:element name="Device_Type" minOccurs="0">
                <xs:simpleType>
                  <xs:restriction base="xs:string">
                    <xs:enumeration value="Decoder"/>
                    <xs:enumeration value="Encoder"/>
                    <xs:enumeration value="Gateway"/>
                    <xs:enumeration value="Router"/>
                    <xs:enumeration value="Camera"/>
                  </xs:restriction>
                </xs:simpleType>
              </xs:element>
              <xs:element name="Device_Class" minOccurs="0">
                <xs:simpleType>
                  <xs:restriction base="xs:string">

```

10

20

30

40


```
    <xs:enumeration value="PC"/>
    <xs:enumeration value="PDA"/>
    <xs:enumeration value="STB"/>
    <xs:enumeration value="Printer"/>
    <xs:enumeration value="MobilePhone"/>
    <xs:enumeration value=""/>
  </xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="Vendor" type="xs:string" minOccurs="0"/>
<xs:element name="Model" type="xs:string" minOccurs="0"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Hardware" minOccurs="0">
  <xs:complexType>
    <xs:sequence maxOccurs="unbounded">
      <xs:element name="CPU" minOccurs="0">
        <xs:complexType>
          <xs:attribute name="Vendor" type="xs:string" use="optional"/>
          <xs:attribute name="Model" type="xs:string" use="required"/>
          <xs:attribute name="Speed" type="xs:integer" use="required"/>
        </xs:complexType>
      </xs:element>
      <xs:element name="Memory" minOccurs="0">
        <xs:complexType>
          <xs:attribute name="Vendor" type="xs:string" use="optional"/>
          <xs:attribute name="Model" type="xs:string" use="required"/>
          <xs:attribute name="Size" type="xs:integer" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
```

10

20

30

40

```

    </xs:complexType>
  </xs:element>
  <xs:element name="Speaker" minOccurs="0">
    <xs:complexType>
      <xs:attribute name="Vendor" type="xs:string" use="optional"/>
      <xs:attribute name="Type" type="xs:string" use="required"/>
    </xs:complexType>
  </xs:element>
  <xs:element name="Keyboard" minOccurs="0">
    <xs:complexType>
      <xs:attribute name="Vendor" type="xs:string" use="optional"/>
      <xs:attribute name="Type" type="xs:string" use="required"/>
    </xs:complexType>
  </xs:element>
  <xs:element name="Screen" minOccurs="0">
    <xs:complexType>
      <xs:sequence maxOccurs="unbounded">
        <xs:element name="Size">
          <xs:complexType>
            <xs:attribute name="horizontal" type="xs:integer" use="requi
red"/>
            <xs:attribute name="vertical" type="xs:integer" use="require
d"/>
          </xs:complexType>
        </xs:element>
        <xs:element name="SizeChar" minOccurs="0">
          <xs:complexType>
            <xs:attribute name="horizontal" type="xs:integer" use="requi
red"/>

```

10

20

30

40

```

        <xs:attribute name="vertical" type="xs:integer" use="require
d"/>
        </xs:complexType>
    </xs:element>
    <xs:element name="BitPerPixel" type="xs:integer" minOccurs="0"
/>
    <xs:element name="PointingResolution" type="xs:string" minOccu
rs="0"/>
    </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Support" minOccurs="0">
    <xs:complexType>
        <xs:sequence minOccurs="0" maxOccurs="unbounded">
            <xs:element name="Color" minOccurs="0">
                <xs:complexType>
                    <xs:attribute name="Capability" type="xs:boolean" use="requi
red"/>
                </xs:complexType>
            </xs:element>
            <xs:element name="TextInput" minOccurs="0">
                <xs:complexType>
                    <xs:attribute name="Capability" type="xs:boolean" use="requi
red"/>
                </xs:complexType>
            </xs:element>
            <xs:element name="ImageInput" minOccurs="0">
                <xs:complexType>
                    <xs:attribute name="Capability" type="xs:boolean" use="requi

```

10

20

30

40

```

red"/>
    </xs:complexType>
  </xs:element>
  <xs:element name="VoiceInput" minOccurs="0">
    <xs:complexType>
      <xs:attribute name="Capability" type="xs:boolean" use="requi

```

```

red"/>

```

10

```

    </xs:complexType>
  </xs:element>
  <xs:element name="SoftKey" minOccurs="0">
    <xs:complexType>
      <xs:attribute name="Capability" type="xs:boolean" use="requi

```

```

red"/>

```

```

    </xs:complexType>
  </xs:element>
  <xs:element name="SoundOutput" minOccurs="0">
    <xs:complexType>
      <xs:attribute name="Capability" type="xs:boolean" use="requi

```

20

```

red"/>

```

```

    </xs:complexType>
  </xs:element>
  </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="AssistHardware" minOccurs="0">
  <xs:complexType>
    <xs:sequence minOccurs="0" maxOccurs="unbounded">
      <xs:element name="SmartCard" minOccurs="0">
        <xs:complexType>

```

30

40

```

    <xs:attribute name="Vendor" type="xs:string" use="optional"/
  )
    <xs:attribute name="Model" type="xs:string" use="required"/>
  </xs:complexType>
</xs:element>
<xs:element name="HardKey" minOccurs="0">
  <xs:complexType>
    <xs:attribute name="type" type="xs:string" use="required"/>
  </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Software" minOccurs="0">
  <xs:complexType>
    <xs:sequence maxOccurs="unbounded">
      <xs:element name="OS" minOccurs="0">
        <xs:complexType>
          <xs:attribute name="Vendor" type="xs:string" use="optional"/>
          <xs:attribute name="Name" type="xs:string" use="required"/>
          <xs:attribute name="Version" type="xs:float" use="required"/>
          <xs:attribute name="SerialNo" type="xs:nonNegativeInteger" use=
"optional"/>
        </xs:complexType>
      </xs:element>
      <xs:element name="VM" minOccurs="0">

```

10

20

30

40

```

<xs:complexType>
  <xs:attribute name="Vendor" type="xs:string" use="required"/>
  <xs:attribute name="Name" type="xs:string" use="required"/>
  <xs:attribute name="Version" type="xs:string" use="required"/>
</xs:complexType>
</xs:element>
<xs:element name="Firmware" minOccurs="0">
  <xs:complexType>
    <xs:attribute name="Vendor" type="xs:string" use="required"/>
    <xs:attribute name="Name" type="xs:string" use="required"/>
    <xs:attribute name="Version" type="xs:string" use="required"/>
  </xs:complexType>
</xs:element>
<xs:element name="Support" minOccurs="0">
  <xs:complexType>
    <xs:sequence minOccurs="0" maxOccurs="unbounded">
      <xs:element name="Download" minOccurs="0">
        <xs:complexType>
          <xs:attribute name="Capability" type="xs:boolean" use="requi
red"/>
        </xs:complexType>
      </xs:element>
      <xs:element name="Browser" minOccurs="0">
        <xs:complexType>
          <xs:attribute name="Name" type="xs:string" use="required"/>
          <xs:attribute name="Version" type="xs:float" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

10

20

30

40

```

        </xs:sequence>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="System" minOccurs="0">
  <xs:complexType>
    <xs:sequence maxOccurs="unbounded">
      <xs:element name="General" minOccurs="0">
        <xs:complexType>
          <xs:sequence minOccurs="0">
            <xs:element name="Modules" minOccurs="0">
              <xs:complexType>
                <xs:attribute name="ModuleID" type="xs:unsignedInt" use="required"/>
                <xs:attribute name="Processing" type="xs:boolean" use="required"/>
              </xs:complexType>
            </xs:element>
            <xs:element name="Components" minOccurs="0">
              <xs:complexType>
                <xs:attribute name="ComponentID" type="xs:unsignedInt" use="required"/>
                <xs:attribute name="Interconnect" type="xs:boolean" use="required"/>
              </xs:complexType>
            </xs:element>
            <xs:element name="Configuration" minOccurs="0">

```

```

    <xs:complexType>
      <xs:attribute name="Option" type="xs:boolean" use="required"
/
    </xs:complexType>
  </xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="IPMP" minOccurs="0">
  <xs:complexType>
    <xs:sequence minOccurs="0">
      <xs:element name="ResidingTools" minOccurs="0" maxOccurs="unbo
unded">
        <xs:complexType>
          <xs:attribute name="ToolID" type="xs:unsignedInt"/>
        </xs:complexType>
      </xs:element>
      <xs:element name="RDDREL" minOccurs="0">
        <xs:complexType>
          <xs:attribute name="Capability" type="xs:boolean" use="requi
red"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

10

20

30

40


```
<xs:element name="CodingCapability" minOccurs="0">
  <xs:complexType>
    <xs:sequence maxOccurs="unbounded">
      <xs:element name="Audio" minOccurs="0">
        <xs:complexType>
          <xs:choice>
            <xs:element name="MPEG">
              <xs:complexType>
                <xs:sequence maxOccurs="unbounded">
                  <xs:element name="MPEG-1" minOccurs="0">
                    <xs:complexType>
                      <xs:sequence>
                        <xs:element name="Layer">
                          <xs:simpleType>
                            <xs:restriction base="xs:string">
                              <xs:enumeration value="I"/>
                              <xs:enumeration value="II"/>
                              <xs:enumeration value="III"/>
                            </xs:restriction>
                          </xs:simpleType>
                        </xs:element>
                      </xs:sequence>
                    </xs:complexType>
                  </xs:element>
                <xs:element name="MPEG-2" minOccurs="0">
                  <xs:complexType>
                    <xs:choice>
                      <xs:element name="LowSamplingRateLayer">
                        <xs:simpleType>
```

10

20

30

40

```

    <xs:restriction base="xs:string">
      <xs:enumeration value="I"/>
      <xs:enumeration value="II"/>
      <xs:enumeration value="III"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="BackwardCompatibleMCLayer">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="I"/>
      <xs:enumeration value="II"/>
      <xs:enumeration value="III"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="AAC">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="LowComplexity"/>
      <xs:enumeration value="MainProfile"/>
      <xs:enumeration value="SamplingRateScaleableProfile"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
</xs:choice>
</xs:complexType>

```

10

20

30

40

```
</xs:element>
<xs:element name="MPEG-4" minOccurs="0">
  <xs:complexType>
    <xs:choice>
      <xs:element name="SyntheticProfile">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="Level1"/>
            <xs:enumeration value="Level2"/>
            <xs:enumeration value="Level3"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="SpeechProfile">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="Level1"/>
            <xs:enumeration value="Level2"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="ScalableProfile">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="Level1"/>
            <xs:enumeration value="Level2"/>
            <xs:enumeration value="Level3"/>
            <xs:enumeration value="Level4"/>
          </xs:restriction>
```

10

20

30

40

```
</xs:simpleType>
</xs:element>
<xs:element name="MainProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level1"/>
      <xs:enumeration value="Level2"/>
      <xs:enumeration value="Level3"/>
      <xs:enumeration value="Level4"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="HighQualityProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level1"/>
      <xs:enumeration value="Level2"/>
      <xs:enumeration value="Level3"/>
      <xs:enumeration value="Level4"/>
      <xs:enumeration value="Level5"/>
      <xs:enumeration value="Level6"/>
      <xs:enumeration value="Level7"/>
      <xs:enumeration value="Level8"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="LowDelayProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
```

10

20

30

40

```
    <xs:enumeration value="Level1"/>
    <xs:enumeration value="Level2"/>
    <xs:enumeration value="Level3"/>
    <xs:enumeration value="Level4"/>
    <xs:enumeration value="Level5"/>
    <xs:enumeration value="Level6"/>
    <xs:enumeration value="Level7"/>
    <xs:enumeration value="Level8"/>
  </xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="NaturalProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level1"/>
      <xs:enumeration value="Level2"/>
      <xs:enumeration value="Level3"/>
      <xs:enumeration value="Level4"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="MobileInternetworkingProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level1"/>
      <xs:enumeration value="Level2"/>
      <xs:enumeration value="Level3"/>
      <xs:enumeration value="Level4"/>
      <xs:enumeration value="Level5"/>
```

10

20

30

40

```
        <xs:enumeration value="Level6"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:element>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="ITU-T">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="G.723"/>
      <xs:enumeration value="G.723.1"/>
      <xs:enumeration value="G.726"/>
      <xs:enumeration value="G.728"/>
      <xs:enumeration value="G.729"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="OtherCoding">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="AC3"/>
      <xs:enumeration value="DTS"/>
      <xs:enumeration value="AMR"/>
      <xs:enumeration value="LPCM"/>
      <xs:enumeration value="ATRAC"/>
```

10

20

30

40

```
    <xs:enumeration value="ATRAC2"/>
    <xs:enumeration value="ATRAC3"/>
  </xs:restriction>
</xs:simpleType>
</xs:element>
</xs:choice>
</xs:complexType>
</xs:element>
<xs:element name="Video" minOccurs="0">
  <xs:complexType>
    <xs:choice>
      <xs:element name="MPEG">
        <xs:complexType>
          <xs:sequence maxOccurs="unbounded">
            <xs:element name="MPEG-1" type="xs:boolean" minOccurs="0"/>
            <xs:element name="MPEG-2" minOccurs="0">
              <xs:complexType>
                <xs:choice>
                  <xs:element name="SimpleProfile">
                    <xs:simpleType>
                      <xs:restriction base="xs:string">
                        <xs:enumeration value="MainLevel"/>
                      </xs:restriction>
                    </xs:simpleType>
                  </xs:element>
                  <xs:element name="MainProfile">
                    <xs:simpleType>
                      <xs:restriction base="xs:string">
                        <xs:enumeration value="LowLevel"/>
                      </xs:restriction>
                    </xs:simpleType>
                  </xs:element>
                </xs:choice>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:choice>
  </xs:complexType>
</xs:element>
```

10

20

30

40

```
    <xs:enumeration value="MainLevel"/>
    <xs:enumeration value="High1440Level"/>
    <xs:enumeration value="HighLevel"/>
  </xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="SNRScalableProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="LowLevel"/>
      <xs:enumeration value="MainLevel"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="SpatialScalableProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="MainLevel"/>
      <xs:enumeration value="High1440Level"/>
      <xs:enumeration value="HighLevel"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="HighProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="MainLevel"/>
      <xs:enumeration value="High1440Level"/>
      <xs:enumeration value="HighLevel"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

10

20

30

40


```
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="MultiviewProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="MainLevel"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="V422Profile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="MainLevel"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
</xs:choice>
</xs:complexType>
</xs:element>
<xs:element name="MPEG-4" minOccurs="0">
  <xs:complexType>
    <xs:choice>
      <xs:element name="SimpleProfile">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="Level0"/>
            <xs:enumeration value="Level1"/>
            <xs:enumeration value="Level2"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
    </xs:choice>
  </xs:complexType>
</xs:element>
```

10

20

30

40

```
    <xs:enumeration value="Level3"/>
  </xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="ScalableProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level1"/>
      <xs:enumeration value="Level2"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="AdvancedSimpleProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level0"/>
      <xs:enumeration value="Level1"/>
      <xs:enumeration value="Level2"/>
      <xs:enumeration value="Level3"/>
      <xs:enumeration value="Level4"/>
      <xs:enumeration value="Level5"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="CoreProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level1"/>
      <xs:enumeration value="Level2"/>
```

10

20

30

40

```
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="CoreScalableProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level1"/>
      <xs:enumeration value="Level2"/>
      <xs:enumeration value="Level3"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="AdvancedCoreProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level1"/>
      <xs:enumeration value="Level2"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="MainProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level2"/>
      <xs:enumeration value="Level3"/>
      <xs:enumeration value="Level4"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

10

20

30

40

```
<xs:element name="NbitProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level2"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="AdvancedRealTimeSimpleProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level1"/>
      <xs:enumeration value="Level2"/>
      <xs:enumeration value="Level3"/>
      <xs:enumeration value="Level4"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="AdvancedCodingEfficiencyProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level1"/>
      <xs:enumeration value="Level2"/>
      <xs:enumeration value="Level3"/>
      <xs:enumeration value="Level4"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="SimpleStudioProfile">
  <xs:simpleType>
```

10

20

30

40

```
<xs:restriction base="xs:string">
  <xs:enumeration value="Level1"/>
  <xs:enumeration value="Level2"/>
  <xs:enumeration value="Level3"/>
  <xs:enumeration value="Level4"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="CoreStudioProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level1"/>
      <xs:enumeration value="Level2"/>
      <xs:enumeration value="Level3"/>
      <xs:enumeration value="Level4"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="FGSProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level0"/>
      <xs:enumeration value="Level1"/>
      <xs:enumeration value="Level2"/>
      <xs:enumeration value="Level3"/>
      <xs:enumeration value="Level4"/>
      <xs:enumeration value="Level5"/>
    </xs:restriction>
  </xs:simpleType>
```

10

20

30

40

```
</xs:element>
<xs:element name="SimpleFaceAnimationProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level1"/>
      <xs:enumeration value="Level2"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="SimpleFBAProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level1"/>
      <xs:enumeration value="Level2"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="BasicAnimatedTextureProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level1"/>
      <xs:enumeration value="Level2"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="ScalableTextureProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level1"/>
```

10

20

30

40

```
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="AdvancedScalableTextureProfile">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="Level1"/>
          <xs:enumeration value="Level2"/>
          <xs:enumeration value="Level3"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
    <xs:element name="HybridProfile">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="Level1"/>
          <xs:enumeration value="Level2"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
  </xs:choice>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="ITU_T">
  <xs:simpleType>
    <xs:restriction base="xs:string">
```

10

20

30

40

```
    <xs:enumeration value="H261"/>
    <xs:enumeration value="H263"/>
  </xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="M-JPEG">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="mjp2Profile"/>
      <xs:enumeration value="mj2sProfile"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="OtherCoding">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="RealVideo"/>
      <xs:enumeration value="WMV"/>
      <xs:enumeration value="QuickTime"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
</xs:choice>
</xs:complexType>
</xs:element>
<xs:element name="ImageGraphics" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="JPEG"/>
```

10

20

30

40


```
<xs:enumeration value="JPEG2000"/>
<xs:enumeration value="TIFF"/>
<xs:enumeration value="GIF"/>
<xs:enumeration value="XBM"/>
<xs:enumeration value="PNG"/>
<xs:enumeration value=""/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="Text" minOccurs="0">
  <xs:complexType>
    <xs:choice>
      <xs:element name="ASCII" type="xs:boolean"/>
      <xs:element name="Non-ASCII">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="HTML"/>
            <xs:enumeration value="ISO-8859-1"/>
            <xs:enumeration value="Shift_JIS"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
    </xs:choice>
  </xs:complexType>
</xs:element>
<xs:element name="ControlTerm" minOccurs="0">
  <xs:complexType>
    <xs:sequence maxOccurs="unbounded">
      <xs:element name="DisplayWinSize" minOccurs="0">
```

10

20

30

40

```

    <xs:complexType>
      <xs:attribute name="Width" type="xs:integer" use="required"/
    )
      <xs:attribute name="Height" type="xs:integer" use="required"
    />
  </xs:complexType>
</xs:element>
<xs:element name="PictureSize" minOccurs="0">
  <xs:complexType>
    <xs:attribute name="Width" type="xs:integer" use="required"/
  )
    <xs:attribute name="Height" type="xs:integer" use="required"
  />
  </xs:complexType>
</xs:element>
<xs:element name="Bitrate" type="xs:integer" minOccurs="0"/>
<xs:element name="BufferSize" type="xs:integer" minOccurs="0"/
)
  <xs:element name="FrameRate" type="xs:integer" minOccurs="0"/>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:schema>

```

10

20

30

40

【0114】

【発明の効果】

定義された仕組みで構築されたマルチメディア・フレームワークは、コンテンツ検索、コンテンツ配信、ビデオ・オン・デマンド、デジタル・ライブラリ・サービス等で用いられる、異なる端末およびネットワーク間でコンテンツの異なるフォーマットに適合できる。

【0115】

定義された構造化 D I A 記述で構築された端末は、異なるフォーマットのコンテンツをア

クセスし、検索し、かつ消費することができる。そして端末は、その有用性を拡張するために異なるアプリケーションおよび異なるネットワーク条件で用いることができる。

【0116】

定義された構造化DIA記述で構築された適合ゲートウェイは、サポートされた端末で利用される要求されたフォーマットに一致するために別のフォーマットにコンテンツを変換すべく、そのDIA記述を有する入力コンテンツをパーズしかつ処理できる。

【0117】

定義された構造化DIA記述で構築されたサーバは、それがそのビジネススケールを拡張するために様々な端末に対してサブできるように、端末によって要求されたフォーマットに一致するために、一つのフォーマットから別のフォーマットにコンテンツを変換するために適合ツールを選択できる。

10

【0118】

本発明は、以下に示す様々な態様をとることができる。本発明の第1態様のユニバーサル・マルチメディア・フレームワーク端末装置のコンテンツ適合方法は、所定の規格に準拠するコンテンツ復号化ツールで端末を構築するステップと、DIA（デジタル・アイテム適合）記述を用いて前記端末を記述するステップと、

前記端末に前記DIA記述を添付するステップと、

コンテンツの一部を検索するために、サーバに前記DIA記述を送り、実行依頼するステップであって、前記サーバが前記DIA記述を構文解析して処理するステップと、

前記端末によって実行依頼された前記DIA記述に一致したフォーマットのコンテンツを選択するステップと、

20

前記端末に前記コンテンツを配信するステップとを含む。

【0119】

本発明の第2態様のユニバーサル・マルチメディア・フレームワーク端末装置のコンテンツ適合方法は、所定の規格に準拠するコンテンツ符号化ツールで端末を構築するステップと、

DIA（デジタル・アイテム適合）記述を用いて前記端末を記述するステップと、

前記端末に前記DIA記述を添付するステップと、

前記端末にDIA記述発生器を実装して、オンラインでDIA記述を生成可能にするステップと、

30

前記端末のネットワーク・プロトコル、他のツール、またはユーザ・プリファレンスに基づいて、ネットワーク条件およびユーザ・プリファレンスを記述したDIA記述をオンラインで生成するステップと、

コンテンツの一部を検索するために、サーバに前記DIA記述を送り、実行依頼するステップであって、前記サーバが前記DIA記述を構文解析して処理するステップと、

前記端末によって実行依頼された前記DIA記述に一致したフォーマットのコンテンツを選択するステップと、

前記端末に前記コンテンツを配信するステップとを含む。

40

【0120】

本発明の第3態様のユニバーサル・マルチメディア・フレームワーク端末装置のコンテンツ適合方法は、所定の規格に準拠するコンテンツ符号化ツールで端末を構築するステップと、

DIA（デジタル・アイテム適合）記述を用いて前記端末を記述するステップと、

前記端末に前記DIA記述を添付するステップと、

前記端末にDIA記述発生器を実装して、オンラインでDIA記述を生成可能にするステップと、

前記端末のネットワーク・プロトコル、他のツール、またはユーザ・プリファレンスに基づいて、ネットワーク条件およびユーザ・プリファレンスを記述したDIA記述をオンラ

50

インで生成するステップと、

前記端末のD I Dパーサ、I P M Pパーサ、R E Lパーサ、R D Dパーサ、およびD I Aパーサを実装して、D I D（デジタル・アイテム・デklarेशन）、I P M P（知的財産権管理および保護）、R E L（権利表現言語）、R D D（権利データ辞書）、およびD I A記述を構文解析するステップと、

そのD I D、I P M P、R E L、R D D、およびD I A記述でD I（デジタル・アイテム）を閲覧するステップと、

前記端末によってD I Dメニューの「チョイス」ノートを選択するステップと、

前記D I DメニューにI P M P、R E L、R D D、およびD I A記述が現されている場合には、前記端末のD I DメニューのI P M P、R E L、R D D、およびD I A記述を構文解析するステップと、

前記D I Dメニューに、適合したD I Aノートが存在するときには、コンテンツを検索するために、前記端末、前記ネットワーク条件、および、前記ユーザ・プリファレンスを記述するのに用いた前記D I A記述を、サーバに実行依頼するステップであって、前記サーバが前記D I A記述を構文解析して処理するステップと、

前記端末によって実行依頼された前記D I A記述に一致したフォーマットであるコンテンツを選択するステップと、

前記端末に前記コンテンツを配信するステップとを含む。

【0121】

本発明の第4態様のユニバーサル・マルチメディア・フレームワーク・ゲートウェイ装置のコンテンツ適合方法は、トランスコーディング機能を有する適合ゲートウェイを構築するステップと、

D I Aパーサを有する適合ゲートウェイを構築するステップと、

端末に添付されたD I A（デジタル・アイテム適合）記述を用いて、前記端末を記述するステップと、

前記適合ゲートウェイによってサポートされると想定されている前記端末の各々について、前記適合ゲートウェイに前記D I A記述を登録するステップと、

前記適合ゲートウェイにおいて、そのD I A記述を有するコンテンツを受け取るステップと、

前記適合ゲートウェイにおいて、前記D I Aパーサにより、前記コンテンツを記述するのに用いた、受け取ったD I A記述を構文解析するステップと、

前記コンテンツについて記述するのに用いた、構文解析された前記D I A記述を、前記端末を記述するのに用いた前記D I A記述と比較するステップと、

上記ステップにおける前記比較結果により示された、符号化フォーマットに関して一致がない場合には、前記コンテンツをトランスコーディングするステップと、

前記ステップによりトランスコーディングされたフォーマットで、前記コンテンツを記述する、新たなD I A記述の組を生成するステップと、

前記トランスコーディングされたフォーマットの前記コンテンツに、前記D I A記述の組を添付するステップと、

前記適合ゲートウェイにより、トランスコーディングされた異なる符号化フォーマットを有するコンテンツを、サポートする前記端末に送信するステップと

を含む。

【0122】

本発明の第5態様のユニバーサル・マルチメディア・フレームワーク・サーバ装置のコンテンツ適合方法は、記憶したマルチメディア・コンテンツでマルチメディア・サーバを構築するステップと、

前記コンテンツの各々に対してD I A記述を生成するステップと、

前記D I A記述を、前記対応するコンテンツに添付するステップと、

前記サーバにD I Aパーサを実装するステップと、

10

20

30

40

50

前記サーバに適合ツールを実装またはプラグインして、異なるマルチメディアフォーマット間でトランスコーディング機能を実行可能にするステップと、
 所定の手段を介して、端末からD I A記述の組を受け取るステップであって、前記D I A記述の組は、前記端末の復号化能力、使用されるネットワークの条件、および、ユーザ・プリファレンスを記述するのに用いられるステップと、

前記サーバによって前記D I A記述の組を構文解析するステップと、

上記ステップにおける構文解析の結果に応じて、前記適合ツールを選択して、トランスコーディングを行い、前記サーバ内で、あるフォーマットから別のフォーマットに変換するステップと、

トランスコーディングされた前記コンテンツを記述する、新たなD I A記述の組を生成するステップと、

メタデータとして、前記新たなD I A記述の組を、前記トランスコーディングされたコンテンツに添付するステップと、

を含む。

【0123】

本発明の第6態様のユニバーサル・マルチメディア・フレームワーク・サーバ装置のコンテンツ適合方法は、記憶したマルチメディア・コンテンツでマルチメディア・サーバを構築するステップと、

前記コンテンツの各々に対してD I A記述を生成するステップと、

前記D I A記述を、前記対応するコンテンツに添付するステップと、

前記サーバにD I Aパーサを実装するステップと、

前記サーバに適合ツールを実装またはプラグインして、異なるマルチメディアフォーマット間でトランスコーディング機能を実行可能にするステップと、

前記サーバにおいて、前記コンテンツを、前記適合ツールを用いて人気がある複数の符号化フォーマットに予めトランスコーディングし、記憶するステップと、

所定の手段を介して、端末からD I A記述の組を受け取るステップであって、前記D I A記述の組は、前記端末の復号化能力、使用されるネットワークの条件、および、ユーザ・プリファレンスを記述するのに用いられるステップと、

前記サーバによって前記D I A記述の組を構文解析するステップと、

前記サーバ内に記憶されたコンテンツであって、前記端末により実行依頼された前記D I A記述を用いて記述された、必要なフォーマットのコンテンツを選択するステップと、
 トランスコーディングされた前記コンテンツを記述する、新たなD I A記述の組を生成するステップと、

メタデータとして、前記新たなD I A記述の組を、前記トランスコーディングされたコンテンツに添付するステップと、

一致した符号化フォーマット、ネットワーク条件およびユーザ・プリファレンスを有する前記コンテンツを、前記端末に配信するステップと、

を含む。

【0124】

本発明の第7態様のユニバーサル・マルチメディア・フレームワーク・サーバ装置のコンテンツ適合方法は、記憶したマルチメディア・コンテンツでマルチメディア・サーバを構築するステップと、

前記コンテンツの各々に対してD I A記述を生成するステップと、

前記D I A記述を、前記対応するコンテンツに添付するステップと、

前記サーバにD I Aパーサを実装するステップと、

前記サーバに適合ツールを実装またはプラグインして、異なるマルチメディアフォーマット間でトランスコーディング機能を実行可能にするステップと、

視聴およびセレクトのため、ユーザに対して、D I DメニューにI P M P、R E L、R D D、およびD I A記述を提供するステップと、

端末を使用する前記ユーザにより、前記D I Dメニューからのチョイスを行うステップと

10

20

30

40

50

、
前記チョイス中に前記端末からD I A記述の組を受け取るステップであって、前記D I A記述の組が、前記端末復号化能力を記述するのに用いられるステップと、
前記チョイス中に、前記端末からD I A記述の組を受け取るステップであって、前記D I A記述の組は、前記ネットワーク条件およびユーザ・プリファレンスを記述するの用に用いられるステップと、
前記サーバにより、前記D I A記述の組を構文解析するステップと、
上記ステップにおける構文解析の結果に応じて、前記適合ツールを選択して、トランスコーディングを行い、前記サーバ内で、あるフォーマットから別のフォーマットに変換するステップと、
トランスコーディングされた前記コンテンツを記述する、新たなD I A記述の組を生成するステップと、
メタデータとして、前記新たなD I A記述の組を、前記トランスコーディングされたコンテンツに添付するステップと、
一致した符号化フォーマット、ネットワーク条件およびユーザ・プリファレンスを有する前記コンテンツを、前記端末に配信するステップと
を含む。

【0125】

本発明の第8態様のユニバーサル・マルチメディア・フレームワーク・サーバ装置のコンテンツ適合方法は、マルチメディア・コンテンツでマルチメディア・サーバを構築するステップと、

前記コンテンツの各々に対してD I A記述を生成するステップと、
前記D I A記述を、前記対応するコンテンツに添付するステップと、
前記サーバにD I Aパーサを実装するステップと、
前記サーバに適合ツールを実装またはプラグインして、異なるマルチメディアフォーマット間でトランスコーディング機能を実行可能にするステップと、
前記サーバにおいて、前記コンテンツを、前記適合ツールを用いて人気がある複数の符号化フォーマットに予めトランスコーディングし、記憶するステップと、
視聴およびセレクトのため、ユーザに対して、D I DメニューにI P M P、R E L、R D D、およびD I A記述を提供するステップと、
端末を使用する前記ユーザにより、前記D I Dメニューからのチョイスを行うステップと

、
前記チョイス中に前記端末からD I A記述の組を受け取るステップであって、前記D I A記述の組が、前記端末復号化能力を記述するのに用いられるステップと、
前記チョイス中に、前記端末からD I A記述の組を受け取るステップであって、前記D I A記述の組は、前記ネットワーク条件およびユーザ・プリファレンスを記述するの用に用いられるステップと、

前記サーバにより、前記D I A記述の組を構文解析するステップと、
前記端末によって実行依頼された前記D I A記述によって記述される前記要求されたフォーマットである、前記サーバで前記記憶されたコンテンツを選択するステップと、
トランスコーディングされた前記コンテンツを記述する、新たなD I A記述の組を生成するステップと、
メタデータとして、前記新たなD I A記述の組を、前記トランスコーディングされたコンテンツに添付するステップと、
一致した符号化フォーマット、ネットワーク条件およびユーザ・プリファレンスを有する前記コンテンツを、前記端末に配信するステップと
を含む。

【0126】

本発明の第9態様のユニバーサル・マルチメディア・フレームワーク端末装置のコンテンツ適合方法は、前記ユニバーサル・マルチメディア・フレームワーク端末装置のコンテン

10

20

30

40

50

ツ適合方法であって、

D I A 記述を用いて前記端末を記述するステップが、

端末の一般的特徴、端末ハードウェア、端末ソフトウェア、端末システム、および、第 1 のレイヤとして、端末復号化能力を含む、階層構造を用いて前記端末を記述するステップと、

主要コンテンツ、ソース復号化レイヤ、システム復号化レイヤ、伝送レイヤと異なる符号化レイヤにおいて、前記 D I A 記述で前記端末復号化能力を記述するステップと、

符号化および復号化パラメータを含む符号化レイヤの各々において、詳細な記述を規定して、前記復号化処理で用いられる異なるパラメータに対して、前記端末によりサポートされる最大範囲を示すステップと、

上述の D I A 記述で用いる制御語であって、共通語の組であり、前記端末と前記サーバとの間で信号を出すのに用いられる制御語を規定するステップと

を含む。

【0127】

本発明の第 10 態様のユニバーサル・マルチメディア・フレームワーク端末装置のコンテンツ適合方法は、前記ユニバーサル・マルチメディア・フレームワーク端末装置のコンテンツ適合方法であって、

D I A 記述を用いて前記端末を記述するステップが、

端末の一般的特徴、端末ハードウェア、端末ソフトウェア、端末システム、および第 1 のレイヤとして端末復号化能力を含む、階層構造を用いて前記端末を記述するステップと、

主要コンテンツ、ソース復号化レイヤ、システム復号化レイヤ、伝送レイヤと異なる符号化レイヤにおいて、前記 D I A 記述で前記端末復号化能力を記述するステップと、

符号化および復号化パラメータを含む符号化レイヤの各々において、詳細な記述を規定して、前記復号化処理で用いられる異なるパラメータに対して、前記端末によりサポートされる最大範囲を示すステップと、

上述の D I A 記述で用いる制御語であって、共通語の組であり、前記端末と前記サーバとの間で信号を出すのに用いられる制御語を規定するステップと

拡張性のために、XML で前記 D I A 記述を表現するステップとを含む。

【0128】

本発明の第 11 態様のユニバーサル・マルチメディア・フレームワーク端末装置のコンテンツ適合方法は、前記ユニバーサル・マルチメディア・フレームワーク端末装置のコンテンツ適合方法であって、

D I A 記述を用いて前記端末を記述するステップが、

端末の一般的特徴、端末ハードウェア、端末ソフトウェア、端末システム、および、第 1 のレイヤとして、端末復号化能力を含む、階層構造を用いて前記端末を記述するステップと、

主要コンテンツ、ソース復号化レイヤ、システム復号化レイヤ、伝送レイヤと異なる符号化レイヤにおいて、前記 D I A 記述で前記端末復号化能力を記述するステップと、

符号化および復号化パラメータを含む符号化レイヤの各々において、詳細な記述を規定して、前記復号化処理で用いられる異なるパラメータに対して、前記端末によりサポートされる最大範囲を示すステップと、

上述の D I A 記述で用いる制御語であって、共通語の組であり、前記端末と前記サーバとの間で信号を出すのに用いられる制御語を規定するステップと

S D L (構文記述言語) の形式で前記 D I A 記述を生成するステップとを含む。

【0129】

本発明の第 12 態様のユニバーサル・マルチメディア・フレームワーク端末装置のコンテンツ適合方法は、前記ユニバーサル・マルチメディア・フレームワーク端末装置のコンテンツ適合方法であって、前記端末内に D I A 記述生成部を実装して、オンラインで D I

10

20

30

40

50

A記述を生成するステップを有し、
 帯域幅、遅延およびパケット損失の制御語を規定することにより、コンテンツフォーマット適合に関連するネットワーク条件を記述し、
 品質、色、大きさ、ストリーミング、ダウンロード、それらの属性によるフィルタリングの制御語を規定することにより、コンテンツ・フォーマット適合に関するユーザ・プリファレンスを記述し、さらに、
 ネットワーク・プロトコル、他のプライベート・ツール、またはユーザ入力から、前記端末においてネットワーク条件を受け取るステップと、
 上述のステップにおいて規定された前記制御語を用いることにより、前記端末において、ネットワーク条件に関する、受け取ったパラメータを、前記D I A記述に変換するステップと、
 前記端末において、ユーザ・インタフェースからのユーザ・プリファレンスを受け取るステップと、
 上述のステップにおいて規定された前記制御語を用いることにより、前記端末において、ユーザ・プリファレンスに関する、受け取ったパラメータを、前記D I A記述に変換するステップと
 を含む。

【0130】

本発明の第13の態様のユニバーサル・マルチメディア・フレームワーク端末装置は、ネットワークに接続されたユニバーサル・マルチメディア・フレームワーク端末装置であって、
 コンテンツ複合化ツールが記録されているメモリ装置と、
 D I A記述を用いて前記端末を記述するD I A記述生成器と、
 D I A記述を前記端末に添付する添付部と、
 一つのコンテンツを検索するために前記D I A記述とコンテンツ要求とをサーバに配信する伝送器と、
 前記サーバで選択され、前記サーバから配信された前記コンテンツを受信する受信機とを備える。

【0131】

本発明の第14の態様のユニバーサル・マルチメディア・フレームワーク・ゲートウェイは、端末とサーバとの間に設けられたユニバーサル・マルチメディア・フレームワーク・ゲートウェイであって、
 メモリ装置と、
 D I A記述を用いて前記端末を記述するD I A記述生成器と、
 D I A記述を前記端末に添付する第1添付部と、
 前記端末に添付されているD I A記述を前記メモリ装置に登録するレジスタと、
 前記端末によるコンテンツ要求に対応するコンテンツを前記サーバから受信する受信機と、
 前記コンテンツに添付され、受信したD I A記述を構文解析するD I Aパーサと、
 構文解析されたD I A記述を前記端末についての前記D I A記述と比較する比較器と、
 前記コンテンツのフォーマットと前記端末のD I A記述に記載されたフォーマットとのミスマッチがある場合には、前記コンテンツのフォーマットを前記端末のD I A記述に記載されたフォーマットにトランスコードするトランスコーダと、
 前記トランスコードされたコンテンツを記述する新たなD I A記述の組を生成する生成器と、
 前記新たなD I A記述の組を前記トランスコードされたコンテンツに第2添付部と、
 前記コンテンツを前記端末に配信する伝送器と、
 を備える。

【0132】

本発明の第15の態様のユニバーサル・マルチメディア・フレームワーク・サーバは、ネ

ットワークを介して端末に接続されているユニバーサル・マルチメディア・フレームワーク・サーバであって、
マルチメディアコンテンツが記録されているメモリ装置と、
各コンテンツについてのD I A 記述を生成するD I A 生成器と
対応するコンテンツに前記D I A 記述を添付する第1添付部と、
端末の復号化能力、ネットワーク条件、及びユーザ・プリファレンスについて記述するために用いられるD I A 記述の組を前記端末から受信する受信機と、
前記受信した前記端末に関するD I A 記述の組を構文解析するD I A パーサと、
前記コンテンツを異なるマルチメディア・フォーマット間でトランスコードする適合ツールと、
前記構文解析した結果に従ってトランスコードして一つのフォーマットから他のフォーマットに変換する一つの適合ツールを選択するセクタと、
前記トランスコードされたコンテンツを記述する新たなD I A 記述の組を生成する生成器と、
前記新たなD I A 記述の組を前記トランスコードされたコンテンツにそのメタデータとして添付する第2添付部と、
前記コンテンツを前記端末に配信する配信部と
を備える。

【図面の簡単な説明】

【図1】従来技術1－異なるタイプの端末を有するユーザへ、一方向でコンテンツを配信する図である。 20

【図2】従来技術2－異なるタイプの端末を有するユーザへ、双方向でコンテンツを検索し、アクセスする図である。

【図3】MPEG-2 I D I A 記述およびMPEG-2 I D I A パーサを有する端末を示す図である。

【図4】メディア・リソースを記述する、各層に対する階層D I A 記述を示す図である。

【図5】端末記述XMLスキーマのアーキテクチャを示す図である。

【図6】端末記述XMLスキーマのアーキテクチャを示す図である。

【図7】メディア・リソースおよび端末に対するD I A 記述を示す図である。

【図8】そのD I A 記述を有する一つのフォーマットからそのD I A 記述を有する別のフォーマットへの適合ゲートウェイ／端末を示す図である。 30

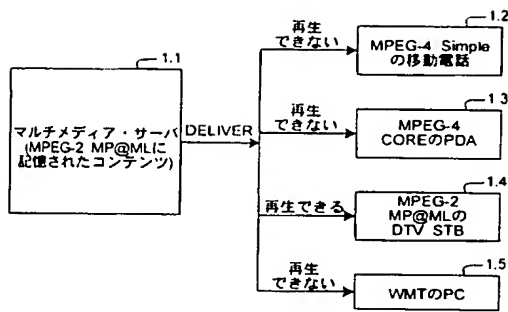
【図9】一方向アプリケーションで用いられるD I A 記述を有するMPEG-2 I 端末を示す図である。

【図10】双方向インタラクティブ・アプリケーションで用いられるD I A 記述を有するMPEG-2 I 端末を示す図である。

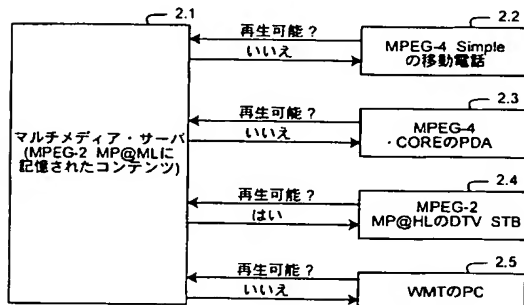
【図11】ストリーミング・コンテンツ配信に対するリアルタイム適合フレームワークを示す図である。

【図12】異なるネットワーク条件およびユーザ・プリファレンスを有する端末への包括的な適合フレームワークを示す図である。

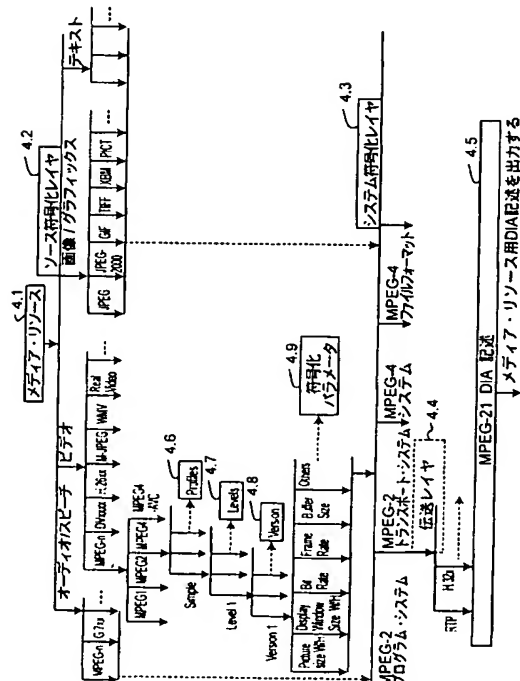
【图 1】



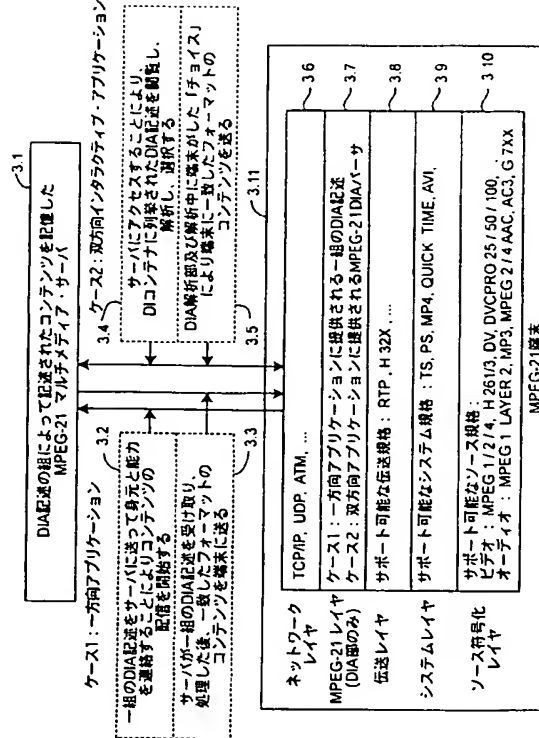
【圖 2】



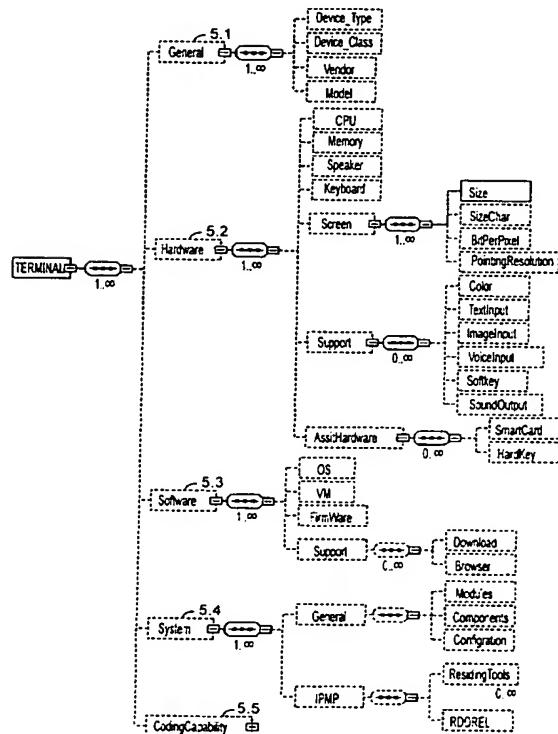
【 図 4 】



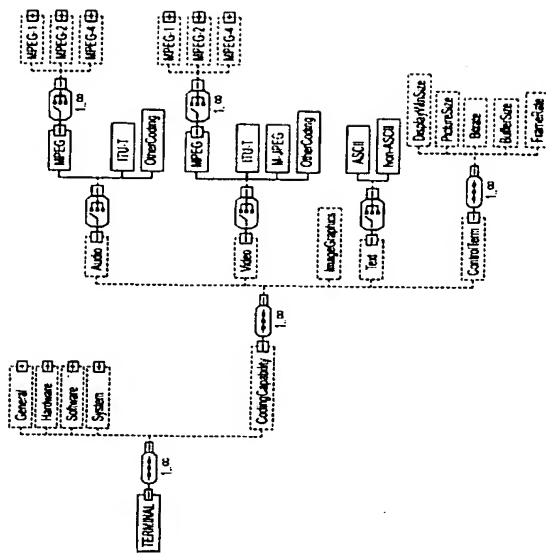
【图 3】



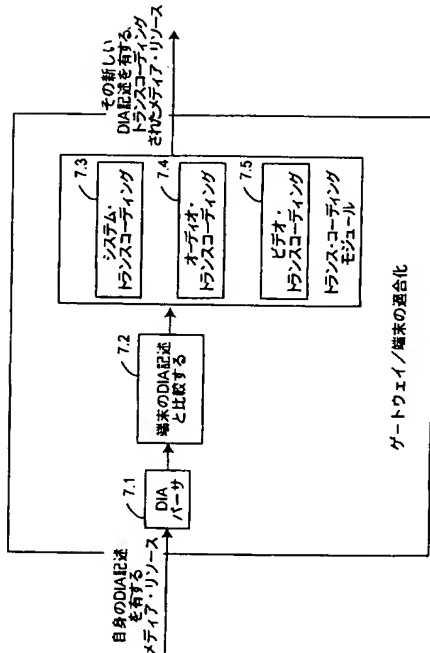
【 図 5 】



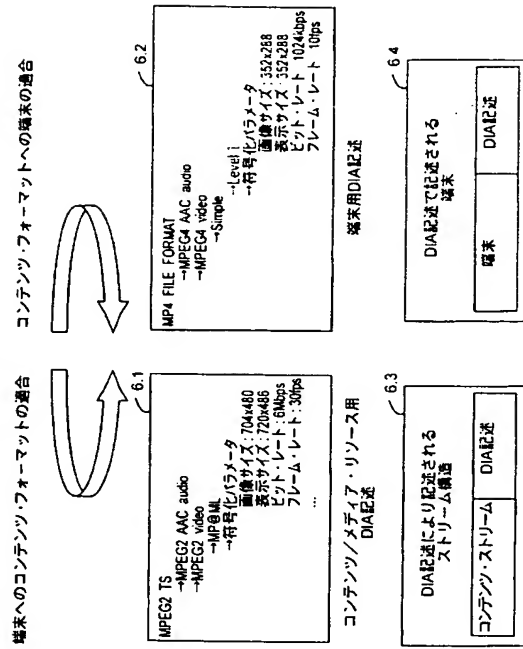
【図 6】



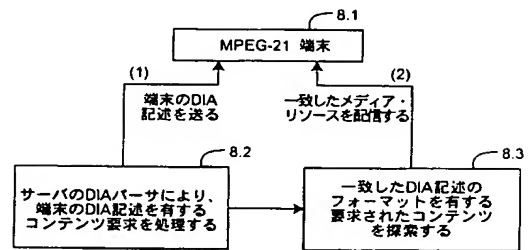
【図 8】



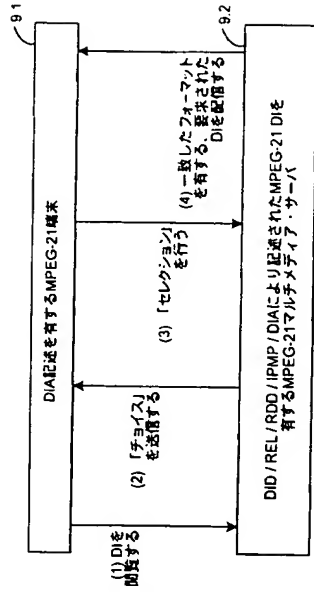
【図 7】



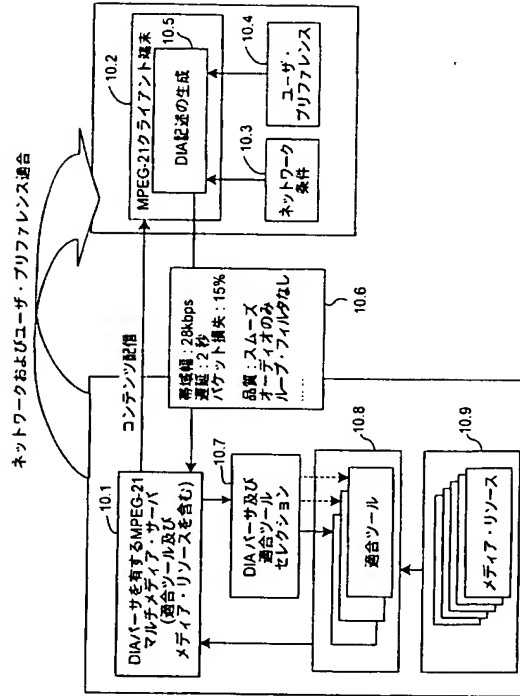
【図 9】



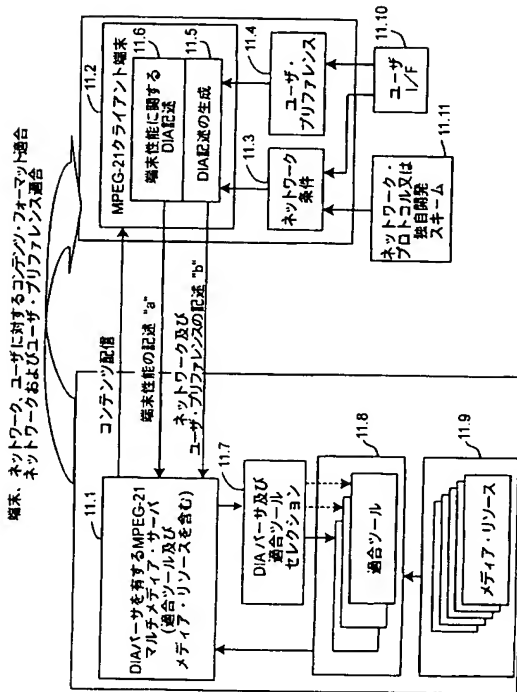
【図10】



【図11】



【図12】



フロントページの続き

(72)発明者 ファング・ゾンヤン

シンガポール 534415 シンガポール、タイ・セン・アベニュー、ブロック 1022、04-3530 番、タイ・セン・インダストリアル・エステイト、パナソニック・シンガポール研究所株式会社内

(72)発明者 ジ・ミン

シンガポール 534415 シンガポール、タイ・セン・アベニュー、ブロック 1022、04-3530 番、タイ・セン・インダストリアル・エステイト、パナソニック・シンガポール研究所株式会社内

(72)発明者 妹尾 孝憲

大阪府門真市大字門真 1006 番地 松下電器産業株式会社内

Fターム(参考) 5B082 AA13 GA02

【外國語明細書】

1 Title of Invention

METHOD OF CONTENT ADAPTATION FOR APPARATUSES OF UNIVERSAL MULTIMEDIA FRAMEWORK TERMINAL, SERVER, AND GATEWAY

2 Claims

1. Method of content adaptation for an Apparatus of Universal Multimedia Framework Terminal, comprising following steps of:

building a terminal with content decoding tools that is compliant to certain standard;

describing the terminal using DIA (Digital Item Adaptation) Descriptions;

attaching the DIA Descriptions to the terminal;

sending and submitting the DIA Descriptions to a server for retrieving a piece of content, whereby the server is parsing and processing the DIA Descriptions;

selecting the content which is in the matched format with the DIA Description submitted by the terminal; and

delivering the content to the terminal.

2. Method of content adaptation for an Apparatus of Universal Multimedia Framework Terminal, comprising the following steps of:

building a terminal with content coding tools that is compliant to certain standard;

describing the terminal using DIA (Digital Item Adaptation) Descriptions;

attaching the DIA Descriptions to the terminal;

implementing a DIA Description Generator in the terminal to be able to generate DIA Descriptions on-line;

generating DIA Descriptions on-line to describe network condition and user preference based on network protocol, other tools, or user preference in the terminal;

sending and submitting the DIA Descriptions to a server for retrieving a piece of content, whereby the server is parsing and processing the DIA Descriptions;

selecting the content which is in the matched format with the DIA Description submitted by the terminal; and
delivering the content to the terminal.

3. Method of content adaptation for an Apparatus of Universal Multimedia Framework Terminal, comprising the following steps of:

building a terminal with content coding tools that is compliant to certain standard;

describing the terminal using DIA (Digital Item Adaptation) Descriptions;

attaching the DIA Descriptions to the terminal;

implementing a DIA Description Generator in the terminal to be able to generate DIA Descriptions on-line;

generating DIA Descriptions on-line to describe network condition and user preference based on network protocol, other tools, or user preference in the terminal;

implementing a DID (Digital Item Declaration) Parser, IPMP (Intellectual Property Management and Protection) Parser, REL (Rights Expression Language) Parser, RDD (Rights Data Dictionary) Parser, and DIA Parser in the terminal to be able to parse DID, IPMP, REL, RDD, and DIA descriptions

browsing a DI (digital Item) with its DID, IPMP, REL, RDD, and DIA Description;

selecting "choice" notes in DID menu by the terminal;

parsing IPMP, REL, RDD, and DIA descriptions in DID menu in the terminal if there is such description appearing in the DID menu;

submitting the DIA Descriptions used to describe the termi

nal, the network condition, and the user preference to a server for retrieving a piece of content when there is such DIA note met in the DID menu, whereby the server is parsing and processing the DIA Descriptions;

selecting the content which is in the matched format with the DIA Description submitted by the terminal; and

delivering the content to the terminal.

4. Method of content adaptation for an Apparatus of Universal Multimedia Framework Gateway, comprising the following steps of:

building an adaptation gateway with transcoding function;

building an adaptation gateway with a DIA Parser;

describing terminal using DIA (Digital Item Adaptation) Descriptions, attached to the terminal;

registering the DIA Descriptions into the adaptation gateway for each of the terminal supposed to be supported by the adaptation gateway;

receiving content with its DIA descriptions in the adaptation gateway;

parsing the received DIA descriptions used to describe the content by the DIA Parser in the adaptation gateway;

comparing the parsed DIA descriptions used to describe for the content with the DIA descriptions used to describe for the terminal ;

transcoding the content if there is mismatched in term of coding formats indicated by the comparing results mentioned in the above step;

generating a new set of DIA descriptions to describe the content in the transcoded coding format done in the above step;

attaching the new set of DIA descriptions to the content in the transcoded coding format; and

sending the contents with different transcoded coding formats by the adaptation gateway to the supported terminal.

5. Method of content adaptation for an Apparatus of Universal Multimedia Framework Server, comprising the following steps of:

building a multimedia server with stored multimedia contents;

creating DIA descriptions for each content;

attaching the DIA descriptions to the corresponding content;

implementing a DIA Parser in the server;

implementing or plugging-in adaptation tools in the server to be able to perform transcoding functions between different multimedia formats;

receiving a set of DIA descriptions from a terminal via any means, wherein the set of DIA descriptions are used to describe for the terminal decoding capability, network condition used, as well as user preference;

parsing the set of DIA descriptions by the server;

selecting the adaptation tools to perform transcoding to convert from one format to another in the server, according to the parsing results mentioned in the above step;

generating a new set of DIA descriptions to describe the transcoded content;

attaching the new set of DIA descriptions to the transcoded content as its metadata;

delivering the content with the matched coding format, network condition, and user preference to the terminal.

6. Method of content adaptation for an Apparatus of Universal Multimedia Framework Server, comprising the following steps of:

building a multimedia server with stored multimedia contents;

creating DIA descriptions for each content;

attaching the DIA descriptions to the corresponding contents;

implementing a DIA Parser in the server;

implementing or plugging-in adaptation tools in the server to be able to perform transcoding functions between different multimedia formats;

pre-transcoding and storing the content in the server to several popular coding formats using the above the adaptation tools;

receiving a set of DIA descriptions from a terminal via any means, wherein the set of DIA descriptions are used to describe for the terminal decoding capability, network condition used, as well as user preference;

parsing the set of DIA descriptions by the server;

selecting the stored content in the server, which is in the required format that is described by the DIA Descriptions submitted by the terminal;

generating a new set of DIA descriptions to describe the transcoded content;

attaching the new set of DIA descriptions to the transcoded content as its metadata;

delivering the content with the matched coding format, network condition, and user preference to the terminal.

7. Method of content adaptation for an Apparatus of Universal Multimedia Framework Server, comprising the following steps of:

building a multimedia server with stored multimedia contents;

creating DIA descriptions for each content;
attaching the DIA descriptions to the corresponding content;
t;
implementing a DIA Parser in the server;
implementing or plugging-in adaptation tools in the server
to be able to perform transcoding functions between different multimedia formats;
providing IPMP, REL, RDD, and DIA descriptions in DID menu
for user to view and select;
making choices from the DID menu by the user who is using
a terminal;
receiving a set of DIA descriptions from the terminal during the choice making, wherein the set of DIA descriptions are used to describe the terminal decoding capability;
receiving a set of DIA descriptions from the terminal during the choice making, wherein the set of DIA descriptions are used to describe the network condition and user preference;
parsing the above the set of DIA descriptions by the server;
selecting the adaptation tools to perform transcoding to convert from one format to another in the server, according to the parsing results mentioned in the above step;
generating a new set of DIA descriptions to describe the transcoded content;
attaching the new set of DIA descriptions to the transcoded content as its metadata;
delivering the content with the matched coding format, network condition, and user preference to the terminal.

8. Method of content adaptation for an Apparatus of Universal

Multimedia Framework Server, comprising the following steps of:

- building a multimedia server with stored multimedia contents;
- creating DIA descriptions for each content;
- attaching the DIA descriptions to the corresponding content;
- implementing a DIA Parser in the server;
- implementing or plugging-in adaptation tools in the server to be able to perform transcoding functions between different multimedia formats;
- pre-transcoding and storing the content in the server to several popular coding formats using the above the adaptation tools;
- providing IPMP, REL, RDD, and DIA descriptions in DID menu for user to view and select;
- making choices from the DID menu by the user who is using a terminal;
- receiving a set of DIA descriptions from the terminal during the choice making, wherein the set of DIA descriptions are used to describe the terminal decoding capability;
- receiving a set of DIA descriptions from the terminal during the choice making, wherein the set of DIA descriptions are used to describe the network condition and user preference;
- parsing the above the set of DIA descriptions by the server;
- selecting the stored content in the server, which is in the required format that is described by the DIA Descriptions submitted by the terminal;
- generating a new set of DIA descriptions to describe the transcoded content;

attaching the new set of DIA descriptions to the transcoded content as its metadata;

delivering the content with the matched coding format, network condition, and user preference to the terminal.

9. Method of content adaptation for an Apparatus of Universal Multimedia Framework Terminal, according to claim 1, further comprising the following steps of:

describing the terminal by using a hierarchy structure, including terminal general feature, terminal hardware, terminal software, terminal system, and terminal decoding capability as the 1st layer;

describing the terminal decoding capability in different coding layers, from primary content, source decoding layer, system decoding layer, transmission layer, to the DIA descriptions;

defining detail descriptions in each coding layer including coding and decoding parameters, to indicate the maximum supporting ranges by the terminal for different parameters used in the decoding processes;

defining controlling terms to use in the DIA descriptions mentioned in the above, which is a set of common terms and used to signal between the terminal and the server.

10. Method of content adaptation for an Apparatus of Universal Multimedia Framework Terminal, according to claim 1, further comprising the following steps of:

describing the terminal by using a hierarchy structure, including terminal general feature, terminal hardware, terminal software, terminal system, and terminal decoding capability as the 1st layer;

describing the terminal decoding capability in different coding layers, from primary content, source decoding layer, system decoding layer, transmission layer, to the DIA descriptions;

defining detail descriptions in each coding layer including coding and decoding parameters, to indicate the maximum supporting ranges by the terminal for different parameters used in the decoding processes;

defining controlling terms to use in the DIA descriptions mentioned in the above, which is a set of common terms and used to signal between the terminal and the server; and

expressing the DIA descriptions in XML to provide extensibility.

11. Method of content adaptation for an Apparatus of Universal Multimedia Framework Terminal, according to claim 1, further comprising the following steps of:

describing the terminal by using a hierarchy structure, including terminal general feature, terminal hardware, terminal software, terminal system, and terminal decoding capability as the 1st layer;

describing the terminal decoding capability in different coding layers, from primary content, source decoding layer, system decoding layer, transmission layer, to the DIA descriptions;

defining detail descriptions in each coding layer including coding and decoding parameters, to indicate the maximum supporting ranges by the terminal for different parameters used in the decoding processes;

defining controlling terms to use in the DIA descriptions mentioned in the above, which is a set of common terms and used to signal between the terminal and the server; and

creating the DIA descriptions in the form of SDL (Syntactic Description Language).

12. Method of content adaptation for an Apparatus of Universal Multimedia Framework Terminal, according to claim 2, wherein netwo

rk condition that is related to content format adaptation is described by defining controlling terms, like bandwidth, delay, and packet loss; and

user preference that is related to content format adaptation is described by defining controlling terms, like quality, color, size, streaming, downloading, filtering with their attributes; further comprising the following steps of:

receiving network condition in the terminal from network protocol, other private tools, or user input;

converting the received parameters on network condition in the terminal into the DIA descriptions by using the controlling terms defined in the above steps;

receiving user preference in the terminal from User Interface; and

converting the received parameters on user preference in the terminal into the DIA descriptions by using the controlling terms defined in the above steps.

13. An apparatus of universal multimedia framework terminal, the terminal connected to a server via a network, comprising:

a memory device in which content decoding tools are stored;

a DIA descriptor for describing the terminal using DIA description;

an attaching element for attaching the DIA descriptions to the terminal;

a transmitter for sending the DIA descriptions and a content requirement to a server for retrieving a piece of content; and

a receiver for receiving the content selected and delivered from the server.

14. An apparatus of universal multimedia framework gateway, the gateway provided between a terminal and a server, comprising:

a memory device;

a DIA descriptor for describing the terminal using DIA description;

a first attaching element for attaching the DIA descriptions to the terminal;

a register for registering a DIA descriptions attached to the terminal into the memory device;

a receiver for receiving a content with its DIA descriptions according to a content requirement of the terminal from the server;

a DIA parser for parsing the received DIA descriptions attached to the content;

a comparator for comparing the parsed DIA descriptions with the DIA descriptions for the terminal;

a transcoder for transcoding the content from one format to another format, if format mismatch is found between the one format of the content and another format described in the DIA descriptions for the terminal;

a generator for generating a new set of DIA descriptions to describe the transcoded content;

a second attaching element for attaching the new set of DIA descriptions to the transcoded content; and

a transmitter for sending the content to the terminal.

15. An apparatus of universal multimedia framework server, the server connected to a terminal via a network, comprising:

a memory device in which multimedia contents are stored;

a creator for creating DIA description for the each content;

a first attaching element for attaching the DIA description to the corresponding content;

a receiver for receiving a set of DIA descriptions from a terminal, the set of DIA descriptions being used to describe for the terminal decoding capability, network condition, as well as user preference ;

a DIA parser for parsing the received DIA description for the terminal;

adaptation tools for transcoding the content between different multimedia formats of contents;

a selector for selecting one adaptation tool to transcode to convert the content from one format to another format according to the parsing results;

a generator for generating a new set of DIA descriptions to describe the transcoded content;

a second attaching element for attaching the new set of DIA descriptions to the transcoded content as its metadata; and

a delivering element for delivering the content to the terminal.

3 Detailed Description of Invention

TECHNICAL FIELD

The present invention relates to a unified multimedia terminal that can access multimedia content in different formats, and relates to a unified multimedia framework that can handle different multimedia terminal, network, usage environment.

BACKGROUND ART

MPEG and other standard bodies have created many standards in video, audio, system, communication protocol, content representation, content packaging, etc., to ease for content transfer and deliver from

one place to another in an efficient way, as well as content storage with big volume in a limited space.

As a result of it existing contents can be found in different formats due to the different source coding methods, MPEG1, MPEG2, MPEG4, DV, DVCPRO, etc, and at the same time devices and terminals are also built in different scope to be compliant with certain standard but not with another. For example, a DTV STB (Set-Top-Box) can decode and play back MPEG-2 compliant content but cannot play back DV content, or MPEG-4 content.

It would be desirable that there has a universal multimedia terminal, which can access and consume any content no matter what format it is. There are a few ways to achieve this.

One way is to build a terminal that has a powerful transcoder function to understand all the different formats and decode them in real-time, then encode the content into the required format in real-time. But the cost would be very high for having such a real-time multifunctional transcoder. It is definitely not advisable to use for portable terminal, even for DTV, or DVD like devices.

Another way is to have such transcoder function in service provider side / content creation side, or in a gateway site, to pre-transcode and store the same content in different formats.

In such case a set of descriptions or metadata to describe content representation format can be created and attached to the corresponding content. By matching a query submitted by a terminal with the stored metadata in a server, the server will decide which content in the specific format will be delivered to the terminal. Here the terminal has to be a compliant terminal, to be able to generate a set of pre-defined descriptions or metadata that the server can recognize.

Besides the set of descriptions to address content represe

ntation format, a set of descriptions to describe terminal, as well as a set of descriptions to describe other usage environment including network condition and user preference, is also required to define. However usage environment is different from content representation format and terminal, and it is changeable and selectable and not fixed with the terminal like terminal decoding capability.

Here this invention is focusing on defining a set of descriptions and the structure among the descriptions, as well as to define a mechanism for a multimedia framework used across different terminals and network.

This invention is to try to solve the following problems:

The same content or media resource in certain format can be accessed, retrieved, and consumed by any multimedia terminal with different network condition and different user preference.

A terminal with different decoding capability and network condition is able to access, retrieve, and consume any content or media resource that is in different data formats.

DISCLOSURE OF INVENTION

By defining a set of descriptions for content representation format, a piece of content or media resource can be described to reflect its coding format and style, where the set of descriptions can be defined based on coding condition according to certain coding standard, and also based on parameters extracted from the bit stream headers.

By defining a set of descriptions for a terminal, the terminal can be described to reflect its conformed decoding capability.

By defining a set of descriptions for a terminal, the terminal can be described to reflect its processing power, memory, operating system, etc. Such description is sent to a server from the terminal before the server can deliver content in the right coding format.

By defining a set of descriptions for network, the network condition which a terminal is connected to, can be described to reflect communication protocol, average bandwidth, delay status, packet loss status, etc.

By defining a set of descriptions for user preference, user's interest and preference can be described to reflect user's choice and selection.

By defining the hierarchy and structure of the above descriptions, a multimedia framework can be built for content delivery to any terminal via any network.

By defining the hierarchy and structure of the above descriptions, a terminal can be built to form a universal multimedia terminal that may have limited decoding capability.

By defining a mechanism used for constructing a multimedia framework, many of the existing and future applications can be served for more users who have different terminals and different network.

Operation of the Invention

On the terminal side:

A MPEG-21 Terminal is built by implementing one of the source coding tools like MPEG-4 Video decoder for simple profile;

A MPEG-21 Terminal is built by implementing one of the system coding tools like MP4 file format;

A MPEG-21 Terminal is built by implementing one of the Transmission tools like RTP;

A MPEG-21 Terminal is built by implementing MPEG-21 DID (Digital Item Declaration), REL (Rights Expression Language), RDD (Rights Data Description), IPMP (Intellectual Property Management and Protection), and DIA (Digital Item Adaptation) Parser;

A MPEG-21 Terminal is built by setting-up a table to describe

ibe the terminal's characteristics such as compliant decoding format including source coding format, system layer coding format, transmission format, where the table can be expressed in XML (Extensible Markup Language 1.0) schema or in SDL (Syntactic Description Language) syntax, where more detail descriptions will be included in each type of coding format, such as bit rate, display window size, buffer size, etc.

On the content server side:

A content or media resource that is in certain data format is pre-transcoded into different data formats with its defined metadata or a set of defined DIA descriptions to describe each type of formats according to MPEG-21 DIA defined structure.

MPEG-21 Digital Item (DI) is generated by constructing a scene structure where the corresponding DID/IPMP/REL/RDD/DIA descriptions are structurally linked together to tell you what is the digital item and where is the media resource, how it is protected, what are the rights for using the DI, and what is the possible terminal and network adaptation listed in form of "choice" for the media resource in order to be delivered and consumed by users. This is used for two-way interactive application.

For one-way application like broadcasting, MPEG-21 Digital Item will be created according to the specific terminal and network condition received from terminal.

It should be noted that the terms and definitions given in the following apply:

Digital Item: a Digital Item is a structured digital object with a standard representation, identification and metadata within the ISO/IEC 21000 framework. This entity is also the fundamental unit of distribution and transaction within this framework

Digital Item Adaptation: a process where a Digital Item is

subject to a resource adaptation engine and/or a descriptor adaptation engine, which together produce an adapted Digital Item.

Content Digital Item: a Digital Item that is used for the delivery of resources.

BEST MODE FOR CARRYING OUT THE INVENTION

The prior art is illustrated in Fig. 1 and Fig. 2 to state the current situation where a content in certain data format cannot be played back by different terminals.

In Fig. 1, it is shown that MPEG-2 MP@ML content is delivered from Multimedia server in module 1.1 to user in one-way, and only DTV STB in module 1.4 can play back. The other terminals in module 1.2, 1.3, and 1.5 cannot play the content.

In Fig. 2, it is shown that MPEG-2 MP@ML content is delivered from Multimedia server in module 2.1 to user in interactive two-way, and only DTV STB in module 2.4 can play back. The other terminals in module 2.2, 2.3, and 2.5 cannot play the content.

From the above examples it is understood that such situation limits the content accessibility by a broad range of terminals and network.

On the other hand, it is not possible for terminal to be built to enable decoding all different formats of content, which is too much costly.

The following embodiments are arranged in the two major sections:

Hierarchy DIA Descriptions for adaptation to different terminals used in Interactive Two-way or One-way applications and real time streaming adaptation to different network;

Interactive Two-way application case

The interactive application is very popular in existing world, such as DTV Interactive data broadcasting, Internet based services, Interactive DVD, Interactive data access by Mobile phone, etc.

Typical model for this two-way application is that server provides a layer structured menu to be viewed by User, User selects what he want to retrieve by sending request to server from a client terminal, and the server processes the request and delivers the requested content to the client terminal.

Here the content is referred to media resource in MPEG-21.

As shown in Fig. 3, MPEG-21 Multimedia Server is illustrated in module 3.1 where content or media resource is stored attached with a set of DIA descriptions.

In module 3.11 a MPEG-21 terminal is shown and it consists of source coding tool in module 3.10, system layer coding tool in module 3.9, transmission protocol in module 3.8, as well as MPEG-21 DIA parser and a set of DIA descriptions in module 3.7. The terminal is connected to TCP/IP, UDP, ATM, or other network in module 3.6.

The case what we are talking here is corresponding to Case 2: Two-way Interactive Application shown in Fig. 3. First of all, the terminal or user will browse a Digital Item, parse the DID/IPMP/REL/RDD and DIA descriptions based on what a user selects in module 3.5. The server delivers the content with matched data format to the terminal according to the choices made by the terminal, in module 3.4.

In the above interactive process, the information received by the server consists of two types: User involved and Non-user involved.

- 1) User involved information: those requests or choices made by the user during the browsing, such as user preference;
- 2) Non-user involved information: the terminal characteristics

tics sent from the terminal automatically when the DIA descriptions are parsed through the DIA Parser in the terminal.

One-way Broadcasting Application Case

In this way, terminal is not supposed to communicate with the server interactively like two-way application.

As shown in Fig. 3 for Case 1: one-way Application, the terminal needs to initiate the content delivery by sending a request with a set of DIA descriptions, to tell the server "what I am" and "what I can do" in module 3.2. The server processes the request and the set of DIA descriptions to transmit the content with the matched format to the terminal, in module 3.3.

There are two types of DIA descriptions. One type is related to terminal characteristics, such as terminal complaint decoding format, terminal process power, etc., and this type of the descriptions is fixed with the terminal and not likely be selected by User; while the other type is related to network condition and user preference which is not fixed with the terminal and it is changeable with different network condition and different user preference.

No matter one-way or two-way applications, those DIA descriptions related to terminal characteristics have to send to the server by terminal not by user. For one-way application they can be submitted to server during communication starting period, while for two-way application they can be submitted to server in the beginning or in the middle way of interactive communication. While those DIA descriptions related to network condition and user preference can be given by user or network protocol that is used in the communication during the delivering process.

Hierarchy Description for Each Coding Layer:

As shown in Fig. 4, a general media resource in module 4.1

can be in the form of text, graphics, still image, audio, video, which are considered as source coding format in module 4.2.

If there is combination of audio and video, system coding is required as shown in module 4.3 to synchronize audio and video, such as MPEG-2 Transport system and Program system which are widely used in the products.

In some case a transmission protocol is required as shown in module 4.4, to achieve certain transmission purpose, such as real time transmission.

Such a media resource is formed using various types of coding format, and a set of descriptions is generated to tell the coding structure of the media resource, as shown in module 4.5, called MPEG-21 DIA Description.

Such descriptions and the layer structure among different layers can be expressed in XML or in MPEG-21 file format in the form of SDL (Syntactic Description Language). Here a XML based schema is used to describe such DIA description as one of the possible solutions (the structure of schema is shown in Fig. 5 and Fig. 6 and the detail schema can be found in the end of this section).

In Fig. 5 and Fig. 6 schema structure, we define an exhaustive terminal description including "General" description (5.1), "Hardware" (5.2), "Software" (5.3), and "System" (5.4) properties. The hierarchy description of terminal coding capability can be seen in "CodingCapability" element (5.5).

In general terminal description, the terminal vendor, model and its type/class such as encoder, decoder, gateway, router, PC, PDA, printer, Mobilephone etc are described.

The hardware and software components are two important elements to denote the terminal capability. The brief description of each

element under these two parts are listed below:

CPU: Vendor, Model, Speed of the device CPU, e.g. Intel Pentium III 1.13GHz;

Memory: Vendor, Model, Size of the device memory, e.g. Intel SDRAM PC133 128M;

Speaker: Vendor, Type of the device speaker, e.g. Yamaha YMF753;

Keyboard: Vendor, Type of the device keyboard, e.g. Toshiba 85-key Ergonomic Keyboard;

Screen: It gives more detail of terminal screen,

"Size": device screen size in unit of pixels, composed of the horizontal and vertical values; "SizeChar": device screen size in units of characters, composed of the horizontal and vertical values and determined by device's standard font; "BitPerPixel": the number of bits of colour or grayscale information per pixel, related to the number of colours or shades of gray the device can display in screen; "PointingResolution": the type of resolution of the pointing accessory supported by device screen.

Hardware Support: indicate whether the terminal supports "Colour" (colour display), "TextInput" (text entry), "ImageInput" (image display), "VoiceInput" (any form of voice input), "SoftKey" (Programme soft keys) and "SoundOutput" (sound output through external speaker); Boolean type is used to denote all these supports, e.g. "yes" for "Colour" support means the device's display support colour";

AssistHardware: some assistant hardware supporting security functionality, "SmartCard", "Hardkey";

OS: Vendor, Name, Version, SerialNumber of the terminal's operating system, e.g. Apple Mac OS, 9.0, 12345678;

VM: Vendor, Name, Version of the virtual machine installed

on the device, e.g. SunJRE1.2;

Firmware: Vendor, Name, Version of the firmware to which the device's low-level software conforms;

Software Support: Indicate whether the terminal supports "Software download", "Browser" with its name and version;

System property is also an important aspect for terminal description.

General description of terminal system including processing "Modules", e.g. extra terminal tools for some special functionalities' performing, interconnection of "Components" and the support of "Configuration" option. For MPEG-21 multimedia framework, the special system property "IPMP" should be provided. The IPMP system description includes residing IPMP_Tools with their ToolID and the support of RDD/REL.

The terminal coding capability is described in a structured mode according to content type (Audio, Video, Image/Graphic, Text) and the respective coding type (e.g. MPEG, ITU). In each coding type, profiles and levels are used for further classifying and extending. Besides these coding formats description, the coding parameters ("ControlTerms") such as Picture Size, Display Window Size, Bit Rate, Frame Rate, Buffer Size, etc are the possible control items to use to describe coding format. The details can be found in the following video coding structuring explanation and the terminal schema description.

The detail structuring for video is also shown as in Fig. 4 to illustrate what are the possible items or parameters required to describe video.

As for MPEG-n video coding format, there are MPEG1 video, MPEG2 video, MPEG4 video, and also MPEG4 AVC (Advance Video Coding). Under each MPEG video coding format, there is profile, level, version to tell the compliant level.

So for a given media resource, what profile, what level, and what version has to be given to just describe its video coding format. It is shown as in module 4.6, 4.7, and 4.8 in Fig. 4, respectively, where Simple profile with Level 1 and Version 1 is indicated for the video coding.

On top of that, coding parameters are also required to be included in the DIA video coding descriptions, which is shown in module 4.9, where Picture Size: W/H (Width and Height), Display Window Size: W/H, Bit Rate, Frame Rate, Buffer Size, and others if there are, are the possible items to use to describe video coding format.

Here video can include primary video in different video formats, like CCIR601 4:2:2, RGB with 8 bits for each colour component, etc..

Video format also covers DVxxx that includes DV (Digital Video) format used in DV camera, DVCPR0-25, DVCPR0-50, DVCPR0-HD. These are the popular format found in the original content and captured by digital camera.

H.26x consists of H.261 and H.263, and mainly used in video conference products;

M-JPEG is motion JPEG;

WMV is Window Media Video used widely in PC world;

Real Video is another type of video format used in the product of Real Network;

In the same way as video, Audio and speech also need to cover different formats: MPEG-n audio like MPEG Layer 2, MP3, MPEG2-AAC, MPEG4-AAC, and G series like G.721/2, G.723, G.726, G.729, etc. Image and graphics covers various types of formats, like JPEG, JPEG-2000, GIF, TIFF, XBM, PICT, etc.

DIA description to describe a content structure and a term

inal is shown in Fig. 7.

In module 6.1, DIA descriptions are listed to describe a media resource, while in module 6.2, DIA descriptions are listed to describe a multimedia terminal.

In the front of a content structure, a set of DIA descriptions is always attached by described the detail coding structure of the content, as shown in module 6.3.

On the other hand, for a given terminal a set of DIA descriptions is always requested to be implemented with the terminal to describe the terminal decoding capability and its processing power, as shown in 6.4.

From Fig. 7, it is clear that two formats in module 6.1 and 6.2 do not match and adaptation is required to enable the same content for the media resource can be consumed by the terminal.

The ideal adaptation is to rely on a real-time Adaptation Gateway that contains transcoding module. Once the adaptation gateway receives such DIA descriptions as listed in module 6.1, it will convert or transcode the coding format to fit for the DIA description as listed in module 6.2 for consuming terminal. In this case, for Source Coding Layer, MPEG-2 AAC is transcoded into MPEG-4 AAC, and MPEG-2 video MP@ML (Main Profile and Main Level) transcoded into MPEG-4 video Simple Profile with Level 1 and Version 1. For System Coding Layer, MPEG-2 Transport System is converted and transcoded into MP4 file format. A downsize process is required here before encoding into MPEG-4 Video format from MPEG-2 video due to the different supported Picture Sizes on two sides indicated in Coding Parameters in the two sets of DIA descriptions.

The adaptation gateway is shown in Fig. 8 to illustrate the above solution. It consists of DIA Parser and transcoding module to convert from one format into another and re-deliver to a terminal.

As shown in Fig. 8, in module 7.1 DIA Parser in an Adaptation Gateway is to process the input DIA descriptions received from a server, and in module 7.2 the processed results after module 7.1 are compared to the Terminal's DIA descriptions that are subscribed by the terminal. Based on the results in module 7.2, transcoding in Source Coding layer and System Coding layer will be performed in module 7.3, 7.4 and 7.5, respectively for audio, video, and system coding format.

Sometimes to save computation time, the conversion or transcoding can be done in bit stream level, and such intelligent transcoding tool can be even implemented in a terminal and the implementation can be different in different situations. These will be left to implementers to consider according to their terminal's process power and applications.

There is another better way to serve for the purpose. The useful working model should be relying on the processing power on server side. More important point is the application scenario where a media resource is pre-encoded and stored in a multimedia server or library. In such case, the same content can be pre-transcoded and stored in different coding formats to match with the popular and widely used cases.

A generic block diagram is shown in Fig. 9 for one of the implementations. A MPEG-21 terminal in module 8.1 with its DIA Descriptions is sending a content request to server with its set of DIA descriptions that describe the terminal complaint coding format as shown in module 6.2 in Fig. 7. The server is processing the received DIA descriptions in module 8.2, and searching for the matched coding format for the requested content in module 8.3. Finally the requested content in the matched coding format is being delivered to the terminal from the server. This case is for One-way application like broadcasting.

Another implementation is shown in Fig. 10, which is for i

interactive application. A terminal in module 9.1 browses a Digital Item that is described with DID/REL/RDD/IPMP/DIA in (1), and the server provides different "choices" to indicate terminal characteristics, network condition, or user preference in (2).

There are two ways for server to acquire about terminal characteristics: once user selects "Terminal Characteristics" displayed to the user in (3) as shown in Fig. 10, the terminal will automatically send its DIA Descriptions to the server, and another way is to let user decide for the terminal characteristics in the case the user knows about their terminal by any means. Finally the content with matched coding format as an adapted media resource is delivered to the terminal in (4).

Here Terminal Characteristics includes its complaint coding capability and also its processing power/memory/operating system, etc.

While the terminal's complaint coding capability is corresponding to Content Representation Format that is used to describe a media resource.

They are the same descriptions but just using different terms for different targeted objects: terminal or content/media resource.

Real time Streaming Adaptation to Different Network:

As explained in the above, there are two different types of DIA Descriptions: Fixed Descriptions to describe Terminal Characteristics which is fixed and cannot be changed, and Changeable Descriptions to describe network condition and user preference, which is not fixed with the terminal, and it can be different as network used is different or changed, or user preference is different or changed.

The former descriptions with their hierarchy structure as well as the mechanism to signal adaptation and transcoding are shown in the above sections, and the later descriptions and the mechanism to signal between a server and a client terminal is described in this section.

It is assumed that the same coding format is used for a me

dia resource and a terminal. Due to different network condition like different bandwidths the media resource needs to be adapted to the changeable condition by any means, including using scalable coding techniques in source coding layer and also application network layer.

There are many ways to change bit rate in real-time to adapt to changeable network. These include SNR scalable coding, adjusting bit rate controller, reducing frame rate, truncating bits for enhancement layer, employing Qof (Quality of Service), etc. All of these are considered as different kinds of adaptation tools, and they can be different in different cases.

In Fig. 11, a real-time adaptation framework is shown to illustrate the streaming case where network condition and user preference is fed back to a server all the time so that the server uses such information to determine what kind of media resource should be delivered to a terminal to match with the suitable bit rate.

As shown in Fig. 11, a MPEG-21 multimedia server is shown in module 10.1 where DIA Parser, Adaptation Tools, and Media Resources are included. A MPEG-21 terminal is shown in module 10.2 with DIA Description Generator in module 10.5 to generate DIA descriptions according to the Network Condition in module 10.3 and User Preference in module 10.4.

In module 10.6 a set of descriptions is listed as an example to illustrate what are the possible items needed to submit to the server.

A DIA Parser in module 10.7 in the server is to process the received DIA descriptions from the client terminal so as to decide to select the appropriate adaptation tool from the Adaptation Tool plug-in in module 10.8. Media Resource in module 10.9 is converted by adaptation tool, to adapt to the requested and suitable network condition and use

r preference that is described by DIA description. Finally the requested media resource is delivered from the server to the client terminal.

A generic adaptation framework is shown in Fig. 12 to illustrate how the adaptation is performed for a given terminal with limited decoding capability, different network condition and user preference.

MPEG-21 Multimedia Server is shown in module 11.1 with DIA Parser. Different adaptation tools are implemented or used as plug-in tool in the server side. Contents are stored in the server as media resources to be retrieved and delivered to user based on request.

MPEG-21 Client Terminal is shown in module 11.2, where DIA Descriptions to describe its decoding capability and processing power in module 11.6, as well as a DIA Description on-line generation module to generate DIA description on network condition and user preference in module 11.5 is included.

During the starting of the communication, the terminal will send the set of DIA descriptions on Terminal Capability to the server, as shown in the arrow of "a", to start the content delivery.

Network Condition in module 11.3 and User Preference in module 11.4 will feed back the actual network condition and user preference to the module 11.5 to generate DIA descriptions on-line anytime if it is requested. The terminal will send the set of DIA descriptions on Network Condition and User Preference to the server, as shown in the arrow of "b".

User Interface is shown in module 11.10 to obtain User Preference. Network Condition can be obtained from Network Protocols, or propriety method in module 11.11, and it is also possible to be given by user via User interface.

As shown in module 11.7, DIA Parser in the server side will process the received DIA descriptions from the terminal and select cer

tain adaptation tools form the server or plug-in tools to convert and transcode the requested media resource into the matched media resource to match with the terminal capability and appropriate networking condition and user preference.

Adaptation tools are shown in module 11.8 on the server side, and it can also be implemented in a middle adaptation gateway as shown in Fig. 8. Media Resources shown in module 11.9 are stored in the server, and it can be pre-transcoded and stored in the server to target on several popular coding formats, to avoid real-time transcoding for many users at the same time.

The detail Terminal Description schema is shown below:

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XML Spy v4.3 U (http://www.xmlspy.com) by Huang Zhongyao
(Panasonic Singapore Laboratories Pte Ltd) -->
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema" elementFormDefault="qualified" attributeFormDefault="unqualified">
  <xs:element name="TERMINAL">
    <xs:annotation>
      <xs:documentation>It give the description of terminal capabilities</xs:documentation>
    </xs:annotation>
    <xs:complexType>
      <xs:sequence maxOccurs="unbounded">
        <xs:element name="General" minOccurs="0">
          <xs:complexType>
            <xs:sequence maxOccurs="unbounded">
              <xs:element name="Device_Type" minOccurs="0">
                <xs:simpleType>
```

```

    <xs:restriction base="xs:string">
      <xs:enumeration value="Decoder"/>
      <xs:enumeration value="Encoder"/>
      <xs:enumeration value="Gateway"/>
      <xs:enumeration value="Router"/>
      <xs:enumeration value="Camera"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="Device_Class" minOccurs="0">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="PC"/>
      <xs:enumeration value="PDA"/>
      <xs:enumeration value="STB"/>
      <xs:enumeration value="Printer"/>
      <xs:enumeration value="MobilePhone"/>
      <xs:enumeration value=""/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="Vendor" type="xs:string" min
Occurs="0"/>
<xs:element name="Model" type="xs:string" min0
ccurs="0"/>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Hardware" minOccurs="0">

```

```

<xs:complexType>
  <xs:sequence maxOccurs="unbounded">
    <xs:element name="CPU" minOccurs="0">
      <xs:complexType>
        <xs:attribute name="Vendor" type="xs:string" use="optional"/>
        <xs:attribute name="Model" type="xs:string" use="required"/>
        <xs:attribute name="Speed" type="xs:integer" use="required"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="Memory" minOccurs="0">
      <xs:complexType>
        <xs:attribute name="Vendor" type="xs:string" use="optional"/>
        <xs:attribute name="Model" type="xs:string" use="required"/>
        <xs:attribute name="Size" type="xs:integer" use="required"/>
      </xs:complexType>
    </xs:element>
    <xs:element name="Speaker" minOccurs="0">
      <xs:complexType>
        <xs:attribute name="Vendor" type="xs:string" use="optional"/>
        <xs:attribute name="Type" type="xs:string" use="required"/>
      </xs:complexType>
    </xs:element>
  </xs:sequence>
</xs:complexType>

```

```

</xs:element>
<xs:element name="Keyboard" minOccurs="0">
  <xs:complexType>
    <xs:attribute name="Vendor" type="xs:string" use="optional"/>
    <xs:attribute name="Type" type="xs:string" use="required"/>
  </xs:complexType>
</xs:element>
<xs:element name="Screen" minOccurs="0">
  <xs:complexType>
    <xs:sequence maxOccurs="unbounded">
      <xs:element name="Size">
        <xs:complexType>
          <xs:attribute name="horizontal" type="xs:integer" use="required"/>
          <xs:attribute name="vertical" type="xs:integer" use="required"/>
        </xs:complexType>
      </xs:element>
      <xs:element name="SizeChar" minOccurs="0">
        <xs:complexType>
          <xs:attribute name="horizontal" type="xs:integer" use="required"/>
          <xs:attribute name="vertical" type="xs:integer" use="required"/>
        </xs:complexType>
      </xs:element>
      <xs:element name="BitPerPixel" type="xs:integer"

```

```

r" minOccurs="0"/>
                                <xs:element name="PointingResolution" type="xs
:string" minOccurs="0"/>
                                </xs:sequence>
                                </xs:complexType>
                                </xs:element>
                                <xs:element name="Support" minOccurs="0">
                                <xs:complexType>
                                <xs:sequence minOccurs="0" maxOccurs="unbounde
d">
                                <xs:element name="Color" minOccurs="0">
                                <xs:complexType>
                                <xs:attribute name="Capability" type="xs:boole
an" use="required"/>
                                </xs:complexType>
                                </xs:element>
                                <xs:element name="TextInput" minOccurs="0">
                                <xs:complexType>
                                <xs:attribute name="Capability" type="xs:boole
an" use="required"/>
                                </xs:complexType>
                                </xs:element>
                                <xs:element name="ImageInput" minOccurs="0">
                                <xs:complexType>
                                <xs:attribute name="Capability" type="xs:boole
an" use="required"/>
                                </xs:complexType>
                                </xs:element>
                                <xs:element name="VoiceInput" minOccurs="0">

```

```

        <xs:complexType>
        <xs:attribute name="Capability" type="xs:boole
an" use="required"/>

        </xs:complexType>
        </xs:element>
        <xs:element name="SoftKey" minOccurs="0">
        <xs:complexType>
        <xs:attribute name="Capability" type="xs:boole
an" use="required"/>

        </xs:complexType>
        </xs:element>
        <xs:element name="SoundOutput" minOccurs="0">
        <xs:complexType>
        <xs:attribute name="Capability" type="xs:boole
an" use="required"/>

        </xs:complexType>
        </xs:element>
        </xs:sequence>
        </xs:complexType>
        </xs:element>
        <xs:element name="AssitHardware" minOccurs="0"
>

        <xs:complexType>
        <xs:sequence minOccurs="0" maxOccurs="unbounde
d">

        <xs:element name="SmartCard" minOccurs="0">
        <xs:complexType>
        <xs:attribute name="Vendor" type="xs:string" u
se="optional"/>

```

```

e="required"/>
<xs:attribute name="Model" type="xs:string" us

</xs:complexType>
</xs:element>
<xs:element name="HardKey" minOccurs="0">
<xs:complexType>
<xs:attribute name="type" type="xs:string" use
="required"/>

</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="Software" minOccurs="0">
<xs:complexType>
<xs:sequence minOccurs="0" maxOccurs="unbounded">
<xs:element name="OS" minOccurs="0">
<xs:complexType>
<xs:attribute name="Vendor" type="xs:string" u
se="optional"/>

<xs:attribute name="Name" type="xs:string" use
="required"/>

<xs:attribute name="Version" type="xs:float" u
se="required"/>

<xs:attribute name="SerialNo" type="xs:nonNega
tiveInteger" use="optional"/>

```



```

</xs:complexType>
</xs:element>
<xs:element name="VM" minOccurs="0">
  <xs:complexType>
    <xs:attribute name="Vendor" type="xs:string" use="required"/>
    <xs:attribute name="Name" type="xs:string" use="required"/>
    <xs:attribute name="Version" type="xs:string" use="required"/>
  </xs:complexType>
</xs:element>
<xs:element name="Firmware" minOccurs="0">
  <xs:complexType>
    <xs:attribute name="Vendor" type="xs:string" use="required"/>
    <xs:attribute name="Name" type="xs:string" use="required"/>
    <xs:attribute name="Version" type="xs:string" use="required"/>
  </xs:complexType>
</xs:element>
<xs:element name="Support" minOccurs="0">
  <xs:complexType>
    <xs:sequence minOccurs="0" maxOccurs="unbounded">
      <xs:element name="Download" minOccurs="0">
        <xs:complexType>
          <xs:attribute name="Capability" type="xs:boolean" use="required"/>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```

an" use="required"/>
    </xs:complexType>
    </xs:element>
    <xs:element name="Browser" minOccurs="0">
      <xs:complexType>
        <xs:attribute name="Name" type="xs:string" use
="required"/>
        <xs:attribute name="Version" type="xs:float" u
se="required"/>
      </xs:complexType>
    </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="System" minOccurs="0">
  <xs:complexType>
    <xs:sequence maxOccurs="unbounded">
      <xs:element name="General" minOccurs="0">
        <xs:complexType>
          <xs:sequence minOccurs="0">
            <xs:element name="Modules" minOccurs="0">
              <xs:complexType>
                <xs:attribute name="ModuleID" type="xs:unsigne
dInt" use="required"/>
                <xs:attribute name="Processing" type="xs:boole
an" use="required"/>
              </xs:complexType>
            </xs:element>
          </xs:sequence>
        </xs:complexType>
      </xs:element>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

```

        </xs:complexType>
        </xs:element>
        <xs:element name="Components" minOccurs="0">
        <xs:complexType>
        <xs:attribute name="ComponentID" type="xs:unsignedInt" use="required"/>
        <xs:attribute name="Interconnect" type="xs:boolean" use="required"/>
        </xs:complexType>
        </xs:element>
        <xs:element name="Configuration" minOccurs="0">
        <xs:complexType>
        <xs:attribute name="Option" type="xs:boolean"
use="required"/>
        </xs:complexType>
        </xs:element>
        </xs:sequence>
        </xs:complexType>
        </xs:element>
        <xs:element name="IPMP" minOccurs="0">
        <xs:complexType>
        <xs:sequence minOccurs="0">
        <xs:element name="ResidingTools" minOccurs="0"
maxOccurs="unbounded">
        <xs:complexType>
        <xs:attribute name="ToolID" type="xs:unsignedInt"/>
        </xs:complexType>
        </xs:element>

```

[illegible]

```
<xs:enumeration value="II"/>
<xs:enumeration value="III"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="MPEG-2" minOccurs="0">
  <xs:complexType>
    <xs:choice>
      <xs:element name="LowSamplingRateLayer">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="I"/>
            <xs:enumeration value="II"/>
            <xs:enumeration value="III"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="BackwardCompatibleMCLayer">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="I"/>
            <xs:enumeration value="II"/>
            <xs:enumeration value="III"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
    </xs:choice>
  </xs:complexType>
</xs:element>
```

```

    <xs:element name="AAC">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="LowComplexity"/>
          <xs:enumeration value="MainProfile"/>
          <xs:enumeration value="SamplingRateScaleablePr
ofile"/>

```

```

          <xs:enumeration value=""/>
        </xs:restriction>
      </xs:simpleType>
    </xs:element>
  </xs:choice>
</xs:complexType>
</xs:element>
<xs:element name="MPEG-4" minOccurs="0">
  <xs:complexType>
    <xs:choice>
      <xs:element name="SyntheticProfile">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="Level1"/>
            <xs:enumeration value="Level2"/>
            <xs:enumeration value="Level3"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="SpeechProfile">
        <xs:simpleType>
          <xs:restriction base="xs:string">

```

```
<xs:enumeration value="Level1"/>
<xs:enumeration value="Level2"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="ScalableProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level1"/>
      <xs:enumeration value="Level2"/>
      <xs:enumeration value="Level3"/>
      <xs:enumeration value="Level4"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="MainProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level1"/>
      <xs:enumeration value="Level2"/>
      <xs:enumeration value="Level3"/>
      <xs:enumeration value="Level4"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="HighQualityProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level1"/>
```

```
<xs:enumeration value="Level2"/>
<xs:enumeration value="Level3"/>
<xs:enumeration value="Level4"/>
<xs:enumeration value="Level5"/>
<xs:enumeration value="Level6"/>
<xs:enumeration value="Level7"/>
<xs:enumeration value="Level8"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="LowDelayProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level1"/>
      <xs:enumeration value="Level2"/>
      <xs:enumeration value="Level3"/>
      <xs:enumeration value="Level4"/>
      <xs:enumeration value="Level5"/>
      <xs:enumeration value="Level6"/>
      <xs:enumeration value="Level7"/>
      <xs:enumeration value="Level8"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="NaturalProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level1"/>
      <xs:enumeration value="Level2"/>
```


”>

```
<xs:enumeration value="Level3"/>
<xs:enumeration value="Level4"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="MobileInternetworkingProfile"

<xs:simpleType>
<xs:restriction base="xs:string">
<xs:enumeration value="Level1"/>
<xs:enumeration value="Level2"/>
<xs:enumeration value="Level3"/>
<xs:enumeration value="Level4"/>
<xs:enumeration value="Level5"/>
<xs:enumeration value="Level6"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="ITU-T">
<xs:simpleType>
<xs:restriction base="xs:string">
<xs:enumeration value="G.723"/>
<xs:enumeration value="G.723.1"/>
```

```
<xs:enumeration value="G.726"/>
<xs:enumeration value="G.728"/>
<xs:enumeration value="G.729"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="OtherCoding">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="AC3"/>
      <xs:enumeration value="DTS"/>
      <xs:enumeration value="AMR"/>
      <xs:enumeration value="LPCM"/>
      <xs:enumeration value="ATRAC"/>
      <xs:enumeration value="ATRAC2"/>
      <xs:enumeration value="ATRAC3"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
</xs:choice>
</xs:complexType>
</xs:element>
<xs:element name="Video" minOccurs="0">
  <xs:complexType>
    <xs:choice>
      <xs:element name="MPEG">
        <xs:complexType>
          <xs:sequence minOccurs="0" maxOccurs="unbounded">
            <xs:element name="MPEG-1" type="xs:boolean" mi
```

minOccurs="0"/>

```
<xs:element name="MPEG-2" minOccurs="0">
  <xs:complexType>
    <xs:choice>
      <xs:element name="SimpleProfile">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="MainLevel"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="MainProfile">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="LowLevel"/>
            <xs:enumeration value="MainLevel"/>
            <xs:enumeration value="High1440Level"/>
            <xs:enumeration value="HighLevel"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="SNRScalableProfile">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="LowLevel"/>
            <xs:enumeration value="MainLevel"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
    </xs:choice>
  </xs:complexType>
</xs:element>
```

```
<xs:element name="SpatialScalableProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="MainLevel"/>
      <xs:enumeration value="High1440Level"/>
      <xs:enumeration value="HighLevel"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

<xs:element name="HighProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="MainLevel"/>
      <xs:enumeration value="High1440Level"/>
      <xs:enumeration value="HighLevel"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

<xs:element name="MultiviewProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="MainLevel"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>

<xs:element name="V422Profile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="MainLevel"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

```
</xs:restriction>
</xs:simpleType>
</xs:element>
</xs:choice>
</xs:complexType>
</xs:element>
<xs:element name="MPEG-4" minOccurs="0">
  <xs:complexType>
    <xs:choice>
      <xs:element name="SimpleProfile">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="Level0"/>
            <xs:enumeration value="Level1"/>
            <xs:enumeration value="Level2"/>
            <xs:enumeration value="Level3"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="ScalableProfile">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="Level1"/>
            <xs:enumeration value="Level2"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:element>
      <xs:element name="AdvancedSimpleProfile">
        <xs:simpleType>
```

```
<xs:restriction base="xs:string">
  <xs:enumeration value="Level0"/>
  <xs:enumeration value="Level1"/>
  <xs:enumeration value="Level2"/>
  <xs:enumeration value="Level3"/>
  <xs:enumeration value="Level4"/>
  <xs:enumeration value="Level5"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="CoreProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level1"/>
      <xs:enumeration value="Level2"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="CoreScalableProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level1"/>
      <xs:enumeration value="Level2"/>
      <xs:enumeration value="Level3"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="AdvancedCoreProfile">
  <xs:simpleType>
```

```

<xs:restriction base="xs:string">
  <xs:enumeration value="Level1"/>
  <xs:enumeration value="Level2"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="MainProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level2"/>
      <xs:enumeration value="Level3"/>
      <xs:enumeration value="Level4"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="NbitProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level2"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="AdvancedRealTimeSimpleProfil

```

e">

```

<xs:simpleType>
  <xs:restriction base="xs:string">
    <xs:enumeration value="Level1"/>
    <xs:enumeration value="Level2"/>
    <xs:enumeration value="Level3"/>
  </xs:restriction>
</xs:simpleType>

```

ile")

```

<xs:enumeration value="Level4"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="AdvancedCodingEfficiencyProf
ile")

<xs:simpleType>
<xs:restriction base="xs:string">
<xs:enumeration value="Level1"/>
<xs:enumeration value="Level2"/>
<xs:enumeration value="Level3"/>
<xs:enumeration value="Level4"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="SimpleStudioProfile">
<xs:simpleType>
<xs:restriction base="xs:string">
<xs:enumeration value="Level1"/>
<xs:enumeration value="Level2"/>
<xs:enumeration value="Level3"/>
<xs:enumeration value="Level4"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="CoreStudioProfile">
<xs:simpleType>
<xs:restriction base="xs:string">
<xs:enumeration value="Level1"/>

```



```
<xs:enumeration value="Level2"/>
<xs:enumeration value="Level3"/>
<xs:enumeration value="Level4"/>
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="FGSProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level0"/>
      <xs:enumeration value="Level1"/>
      <xs:enumeration value="Level2"/>
      <xs:enumeration value="Level3"/>
      <xs:enumeration value="Level4"/>
      <xs:enumeration value="Level5"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="SimpleFaceAnimationProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level1"/>
      <xs:enumeration value="Level2"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="SimpleFBAPProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
```

```

    <xs:enumeration value="Level1"/>
    <xs:enumeration value="Level2"/>
  </xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="BasicAnimatedTextureProfile"

```

```

  >

```

```

    <xs:simpleType>
    <xs:restriction base="xs:string">
    <xs:enumeration value="Level1"/>
    <xs:enumeration value="Level2"/>
    </xs:restriction>
    </xs:simpleType>
  </xs:element>
  <xs:element name="ScalableTextureProfile">
  <xs:simpleType>
  <xs:restriction base="xs:string">
  <xs:enumeration value="Level1"/>
  </xs:restriction>
  </xs:simpleType>
  </xs:element>
  <xs:element name="AdvancedScalableTextureProfi
le">

```

```

    <xs:simpleType>
    <xs:restriction base="xs:string">
    <xs:enumeration value="Level1"/>
    <xs:enumeration value="Level2"/>
    <xs:enumeration value="Level3"/>
    </xs:restriction>

```

```
</xs:simpleType>
</xs:element>
<xs:element name="HybridProfile">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="Level1"/>
      <xs:enumeration value="Level2"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
</xs:choice>
</xs:complexType>
</xs:element>
</xs:sequence>
</xs:complexType>
</xs:element>
<xs:element name="ITU_T">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="H261"/>
      <xs:enumeration value="H263"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
<xs:element name="M-JPEG">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="mjp2Profile"/>
      <xs:enumeration value="mj2sProfile"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

```
</xs:restriction>
</xs:simpleType>
</xs:element>
<xs:element name="OtherCoding">
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="RealVideo"/>
      <xs:enumeration value="WMV"/>
      <xs:enumeration value="QuickTime"/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
</xs:choice>
</xs:complexType>
</xs:element>
<xs:element name="ImageGraphics" minOccurs="0">
```

>

```
  <xs:simpleType>
    <xs:restriction base="xs:string">
      <xs:enumeration value="JPEG"/>
      <xs:enumeration value="JPEG2000"/>
      <xs:enumeration value="TIFF"/>
      <xs:enumeration value="GIF"/>
      <xs:enumeration value="XBM"/>
      <xs:enumeration value="PNG"/>
      <xs:enumeration value=""/>
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

29)

```

        <xs:complexType>
        <xs:attribute name="Width" type="xs:integer" u
se="required"/>
        <xs:attribute name="Height" type="xs:integer"
use="required"/>
    </xs:complexType>
</xs:element>
<xs:element name="PictureSize" minOccurs="0">

```

```

        <xs:complexType>
        <xs:attribute name="Width" type="xs:integer" u
se="required"/>
        <xs:attribute name="Height" type="xs:integer"
use="required"/>
        </xs:complexType>
        </xs:element>
        <xs:element name="Bitrate" type="xs:integer" m
inOccurs="0"/>
        <xs:element name="BufferSize" type="xs:integer
" minOccurs="0"/>
        <xs:element name="FrameRate" type="xs:integer"
minOccurs="0"/>
        </xs:sequence>
        </xs:complexType>
        </xs:element>
        </xs:sequence>
        </xs:complexType>
        </xs:element>
        </xs:sequence>
        </xs:complexType>
        </xs:element>
        </xs:sequence>
        </xs:complexType>
        </xs:element>
</xs:schema>

```

EFFECTS OF INVENTION

A multimedia framework built with the defined mechanism is able to adapt to different formats of content across different terminals and networks, which will be used in content retrieval, content delivery, Video-on-Demand, digital library service, etc.

A terminal built with the defined structured DIA descriptions is able to access, retrieve, and consume content in different formats, and it is able to use in different applications and different network conditions to expand its usefulness.

An adaptation gateway built with the defined structured DIA descriptions is able to parse and process the input content with its DIA descriptions, to convert the content into another format to match with the required format used in the supported terminals.

A server built with the defined structured DIA descriptions is able to select adaptation tools to convert content from one into another format, to match with the required format by a terminal, so that it can serve for variety of terminals to expand its business scale.

A mechanism is introduced to use such DIA Descriptions in real-time streaming content delivery applications where both terminal and server are implemented with such DIA Descriptions Generation and DIA Parser.

The invention can have the following structures viewed from various aspects. According to the first, method of content adaptation for an Apparatus of Universal Multimedia Framework Terminal, including following steps of:

- building a terminal with content decoding tools that is compliant to certain standard;

- describing the terminal using DIA (Digital Item Adaptation) Descriptions;

- attaching the DIA Descriptions to the terminal;

- sending and submitting the DIA Descriptions to a server for retrieving a piece of content, whereby the server is parsing and processing the DIA Descriptions;

- selecting the content which is in the matched format with

the DIA Description submitted by the terminal; and
delivering the content to the terminal.

According to the second, method of content adaptation for an Apparatus of Universal Multimedia Framework Terminal, including the following steps of:

building a terminal with content coding tools that is compliant to certain standard;

describing the terminal using DIA (Digital Item Adaptation) Descriptions;

attaching the DIA Descriptions to the terminal;

implementing a DIA Description Generator in the terminal to be able to generate DIA Descriptions on-line;

generating DIA Descriptions on-line to describe network condition and user preference based on network protocol, other tools, or user preference in the terminal;

sending and submitting the DIA Descriptions to a server for retrieving a piece of content, whereby the server is parsing and processing the DIA Descriptions;

selecting the content which is in the matched format with the DIA Description submitted by the terminal; and

delivering the content to the terminal.

According to the third, method of content adaptation for an Apparatus of Universal Multimedia Framework Terminal, including the following steps of:

building a terminal with content coding tools that is compliant to certain standard;

describing the terminal using DIA (Digital Item Adaptation) Descriptions;

attaching the DIA Descriptions to the terminal;

implementing a DIA Description Generator in the terminal to be able to generate DIA Descriptions on-line;

generating DIA Descriptions on-line to describe network condition and user preference based on network protocol, other tools, or user preference in the terminal;

implementing a DID (Digital Item Declaration) Parser, IPMP (Intellectual Property Management and Protection) Parser, REL (Rights Expression Language) Parser, RDD (Rights Data Dictionary) Parser, and DIA Parser in the terminal to be able to parse DID, IPMP, REL, RDD, and DIA descriptions

browsing a DI (digital Item) with its DID, IPMP, REL, RDD, and DIA Description;

selecting "choice" notes in DID menu by the terminal;

parsing IPMP, REL, RDD, and DIA descriptions in DID menu in the terminal if there is such description appearing in the DID menu;

submitting the DIA Descriptions used to describe the terminal, the network condition, and the user preference to a server for retrieving a piece of content when there is such DIA note met in the DID menu, whereby the server is parsing and processing the DIA Descriptions;

selecting the content which is in the matched format with the DIA Description submitted by the terminal; and

delivering the content to the terminal.

According to the fourth, method of content adaptation for an Apparatus of Universal Multimedia Framework Gateway, including the following steps of:

building an adaptation gateway with transcoding function;

building an adaptation gateway with a DIA Parser;

describing terminal using DIA (Digital Item Adaptation) Descriptions, attached to the terminal;

registering the DIA Descriptions into the adaptation gateway for each of the terminal supposed to be supported by the adaptation gateway;

receiving content with its DIA descriptions in the adaptation gateway;

parsing the received DIA descriptions used to describe the content by the DIA Parser in the adaptation gateway;

comparing the parsed DIA descriptions used to describe for the content with the DIA descriptions used to describe for the terminal ;

transcoding the content if there is mismatched in term of coding formats indicated by the comparing results mentioned in the above step;

generating a new set of DIA descriptions to describe the content in the transcoded coding format done in the above step;

attaching the new set of DIA descriptions to the content in the transcoded coding format; and

sending the contents with different transcoded coding formats by the adaptation gateway to the supported terminal.

According to the fifth, method of content adaptation for an Apparatus of Universal Multimedia Framework Server, including the following steps of:

building a multimedia server with stored multimedia contents;

creating DIA descriptions for each content;

attaching the DIA descriptions to the corresponding content;

implementing a DIA Parser in the server;

implementing or plugging-in adaptation tools in the server

to be able to perform transcoding functions between different multimedia formats;

receiving a set of DIA descriptions from a terminal via any means, wherein the set of DIA descriptions are used to describe for the terminal decoding capability, network condition used, as well as user preference;

parsing the set of DIA descriptions by the server;

selecting the adaptation tools to perform transcoding to convert from one format to another in the server, according to the parsing results mentioned in the above step;

generating a new set of DIA descriptions to describe the transcoded content;

attaching the new set of DIA descriptions to the transcoded content as its metadata;

delivering the content with the matched coding format, network condition, and user preference to the terminal.

According to the sixth, method of content adaptation for an Apparatus of Universal Multimedia Framework Server, including the following steps of:

building a multimedia server with stored multimedia contents;

creating DIA descriptions for each content;

attaching the DIA descriptions to the corresponding content;

implementing a DIA Parser in the server;

implementing or plugging-in adaptation tools in the server to be able to perform transcoding functions between different multimedia formats;

pre-transcoding and storing the content in the server to

several popular coding formats using the above the adaptation tools;

receiving a set of DIA descriptions from a terminal via any means, wherein the set of DIA descriptions are used to describe for the terminal decoding capability, network condition used, as well as user preference;

parsing the set of DIA descriptions by the server;

selecting the stored content in the server, which is in the required format that is described by the DIA Descriptions submitted by the terminal;

generating a new set of DIA descriptions to describe the transcoded content;

attaching the new set of DIA descriptions to the transcoded content as its metadata;

delivering the content with the matched coding format, network condition, and user preference to the terminal.

According to the seventh, method of content adaptation for an Apparatus of Universal Multimedia Framework Server, including the following steps of:

building a multimedia server with stored multimedia contents;

creating DIA descriptions for each content;

attaching the DIA descriptions to the corresponding contents;

implementing a DIA Parser in the server;

implementing or plugging-in adaptation tools in the server to be able to perform transcoding functions between different multimedia formats;

providing IPMP, REL, RDD, and DIA descriptions in DID menu for user to view and select;

making choices from the DID menu by the user who is using a terminal;

receiving a set of DIA descriptions from the terminal during the choice making, wherein the set of DIA descriptions are used to describe the terminal decoding capability;

receiving a set of DIA descriptions from the terminal during the choice making, wherein the set of DIA descriptions are used to describe the network condition and user preference;

parsing the above the set of DIA descriptions by the server;

selecting the adaptation tools to perform transcoding to convert from one format to another in the server, according to the parsing results mentioned in the above step;

generating a new set of DIA descriptions to describe the transcoded content;

attaching the new set of DIA descriptions to the transcoded content as its metadata;

delivering the content with the matched coding format, network condition, and user preference to the terminal.

According to the eighth, method of content adaptation for an Apparatus of Universal Multimedia Framework Server, including the following steps of:

building a multimedia server with stored multimedia contents;

creating DIA descriptions for each content;

attaching the DIA descriptions to the corresponding content;

implementing a DIA Parser in the server;

implementing or plugging-in adaptation tools in the server

to be able to perform transcoding functions between different multimedia formats;

pre-transcoding and storing the content in the server to several popular coding formats using the above the adaptation tools;

providing IPMP, REL, RDD, and DIA descriptions in DID menu for user to view and select;

making choices from the DID menu by the user who is using a terminal;

receiving a set of DIA descriptions from the terminal during the choice making, wherein the set of DIA descriptions are used to describe the terminal decoding capability;

receiving a set of DIA descriptions from the terminal during the choice making, wherein the set of DIA descriptions are used to describe the network condition and user preference;

parsing the above the set of DIA descriptions by the server;

selecting the stored content in the server, which is in the required format that is described by the DIA Descriptions submitted by the terminal;

generating a new set of DIA descriptions to describe the transcoded content;

attaching the new set of DIA descriptions to the transcoded content as its metadata;

delivering the content with the matched coding format, network condition, and user preference to the terminal.

According to the ninth, method of content adaptation for an Apparatus of Universal Multimedia Framework Terminal, according to the above, further including the following steps of:

describing the terminal by using a hierarchy structure, in

cluding terminal general feature, terminal hardware, terminal software, terminal system, and terminal decoding capability as the 1st layer;

describing the terminal decoding capability in different coding layers, from primary content, source decoding layer, system decoding layer, transmission layer, to the DIA descriptions;

defining detail descriptions in each coding layer including coding and decoding parameters, to indicate the maximum supporting ranges by the terminal for different parameters used in the decoding processes;

defining controlling terms to use in the DIA descriptions mentioned in the above, which is a set of common terms and used to signal between the terminal and the server.

According to the tenth, method of content adaptation for an Apparatus of Universal Multimedia Framework Terminal, according to the above, further including the following steps of:

describing the terminal by using a hierarchy structure, including terminal general feature, terminal hardware, terminal software, terminal system, and terminal decoding capability as the 1st layer;

describing the terminal decoding capability in different coding layers, from primary content, source decoding layer, system decoding layer, transmission layer, to the DIA descriptions;

defining detail descriptions in each coding layer including coding and decoding parameters, to indicate the maximum supporting ranges by the terminal for different parameters used in the decoding processes;

defining controlling terms to use in the DIA descriptions mentioned in the above, which is a set of common terms and used to signal between the terminal and the server; and

expressing the DIA descriptions in XML to provide extensibility

ility.

According to the eleventh, method of content adaptation for an Apparatus of Universal Multimedia Framework Terminal, according to the above, further including the following steps of:

- describing the terminal by using a hierarchy structure, including terminal general feature, terminal hardware, terminal software, terminal system, and terminal decoding capability as the 1st layer;

- describing the terminal decoding capability in different coding layers, from primary content, source decoding layer, system decoding layer, transmission layer, to the DIA descriptions;

- defining detail descriptions in each coding layer including coding and decoding parameters, to indicate the maximum supporting ranges by the terminal for different parameters used in the decoding processes;

- defining controlling terms to use in the DIA descriptions mentioned in the above, which is a set of common terms and used to signal between the terminal and the server; and

- creating the DIA descriptions in the form of SDL (Syntactic Description Language).

According to the twelfth, method of content adaptation for an Apparatus of Universal Multimedia Framework Terminal, according to the above method, wherein network condition that is related to content format adaptation is described by defining controlling terms, like bandwidth, delay, and packet loss; and

user preference that is related to content format adaptation is described by defining controlling terms, like quality, color, size, streaming, downloading, filtering with their attributes; further including the following steps of:

- receiving network condition in the terminal from network p

rotocol, other private tools, or user input;

converting the received parameters on network condition in the terminal into the DIA descriptions by using the controlling terms defined in the above steps;

receiving user preference in the terminal from User Interface; and

converting the received parameters on user preference in the terminal into the DIA descriptions by using the controlling terms defined in the above steps.

According to the thirteenth, an apparatus of universal multimedia framework terminal, the terminal connected to a server via a network, including:

a memory device in which content decoding tools are stored ;

a DIA descriptor for describing the terminal using DIA description;

an attaching element for attaching the DIA descriptions to the terminal;

a transmitter for sending the DIA descriptions and a content requirement to a server for retrieving a piece of content; and

a receiver for receiving the content selected and delivered from the server.

According to the fourteenth, an apparatus of universal multimedia framework gateway, the gateway provided between a terminal and a server, including:

a memory device;

a DIA descriptor for describing the terminal using DIA description;

a first attaching element for attaching the DIA description

ns to the terminal;

- a register for registering a DIA descriptions attached to the terminal into the memory device;

- a receiver for receiving a content with its DIA descriptions according to a content requirement of the terminal from the server;

- a DIA parser for parsing the received DIA descriptions attached to the content;

- a comparator for comparing the parsed DIA descriptions with the DIA descriptions for the terminal;

- a transcoder for transcoding the content from one format to another format, if format mismatch is found between the one format of the content and another format described in the DIA descriptions for the terminal;

- a generator for generating a new set of DIA descriptions to describe the transcoded content;

- a second attaching element for attaching the new set of DIA descriptions to the transcoded content; and

- a transmitter for sending the content to the terminal.

According to the fifteenth, an apparatus of universal multimedia framework server, the server connected to a terminal via a network, including:

- a memory device in which multimedia contents are stored;

- a creator for creating DIA description for the each content;

- a first attaching element for attaching the DIA description to the corresponding content;

- a receiver for receiving a set of DIA descriptions from a terminal, the set of DIA descriptions being used to describe for the terminal decoding capability, network condition, as well as user preference

;

a DIA parser for parsing the received DIA description for the terminal;

adaptation tools for transcoding the content between different multimedia formats of contents;

a selector for selecting one adaptation tool to transcode to convert the content from one format to another format according to the parsing results;

a generator for generating a new set of DIA descriptions to describe the transcoded content;

a second attaching element for attaching the new set of DIA descriptions to the transcoded content as its metadata; and

a delivering element for delivering the content to the terminal.

Although the present invention has been described in connection with the preferred embodiments thereof with reference to the accompanying drawings, it is to be noted that various changes and modifications are apparent to those skilled in the art. Such changes and modifications are to be understood as included within the scope of the present invention as defined by the appended claims, unless they depart therefrom.

4 Brief Description of Drawings

The present invention will become readily understood from the following description of preferred embodiments thereof made with reference to the accompanying drawings, in which like parts are designated by like reference numeral, and in which:

Fig. 1 shows the Prior Art 1- One-way Content Delivery to Users with Different Types of Terminals;

Fig. 2 shows Prior Art 2 - Two-way Content Retrieval and Accessing by Users with Different Types of Terminals;

Fig. 3 shows A Terminal with MPEG-21 DIA Descriptions and MPEG-21 DIA Parser;

Fig. 4 shows Hierarchy DIA Descriptions for Each Layer to Describe a Media Resource;

Fig. 5 shows an Architecture of Terminal Description XML Schema;

Fig. 6 shows the Architecture of Terminal Description XML Schema continued from Fig. 5;

Fig. 7 shows DIA Descriptions for a Media Resource and a Terminal;

Fig. 8 shows Illustration of an Adaptation Gateway from one format with its DIA Descriptions to another format with its DIA descriptions;

Fig. 9 shows Illustration of MPEG-21 Terminal with DIA Description used in one-way Application;

Fig. 10 shows Illustration of MPEG-21 Terminal with DIA Description used in Interactive two-way Application;

Fig. 11 shows Real-Time Adaptation Framework for Streaming Content Delivery; and

Fig. 12 shows a Generic Adaptation Framework to a Terminal with Different Network Condition and User Preference.

1 Abstract

Method of content adaptation for an Apparatus of Universal Multimedia Framework Terminal is provided. The method includes following steps of: building a terminal with content decoding tools that is compliant to certain standard; describing the terminal using DIA (Digital Item Adaptation) Descriptions; attaching the DIA Descriptions to the terminal; sending and submitting the DIA Descriptions to a server for retrieving a piece of content, whereby the server is parsing and processing the DIA Descriptions; selecting the content which is in the matched format with the DIA Description submitted by the terminal; and delivering the content to the terminal.

2 Representative Drawing Fig. 3

Fig 1

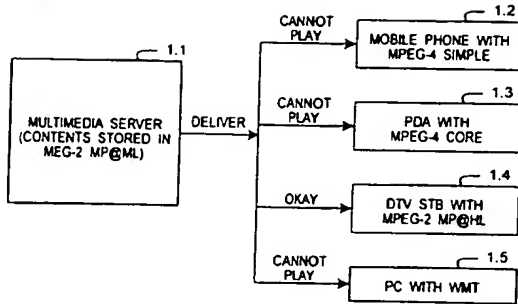


Fig 2

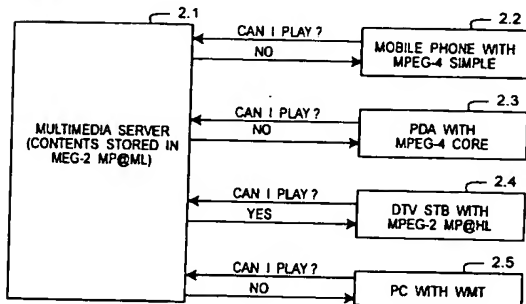


Fig 3

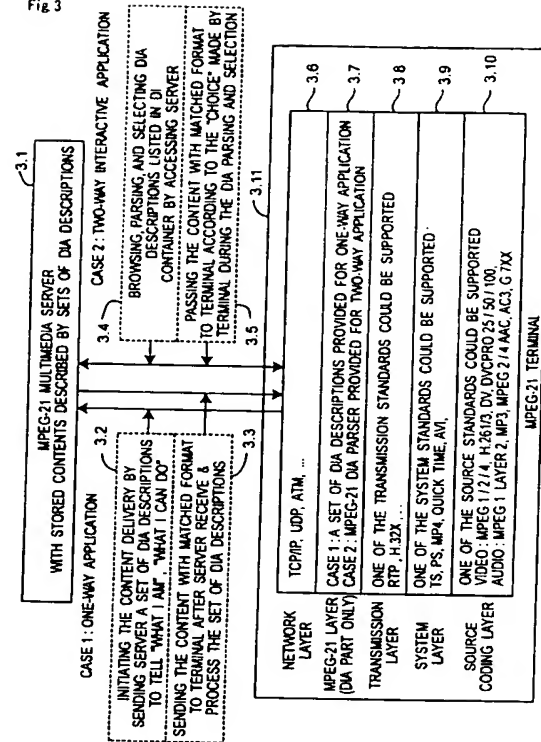


Fig 4

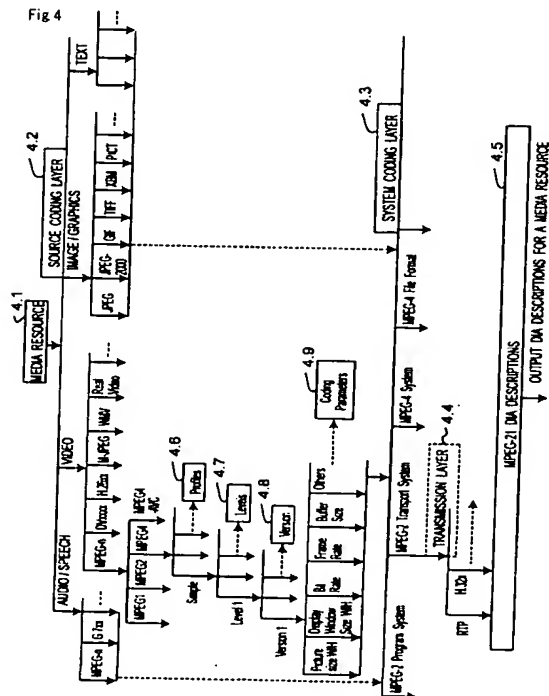


Fig 5

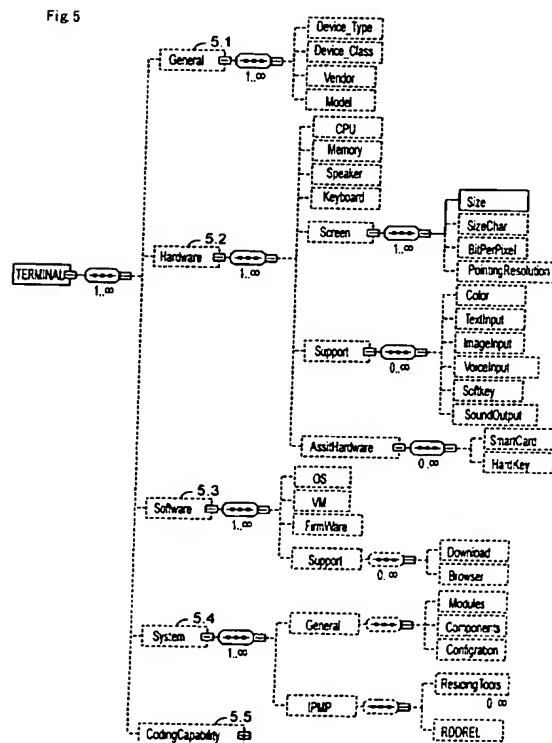


Fig 6

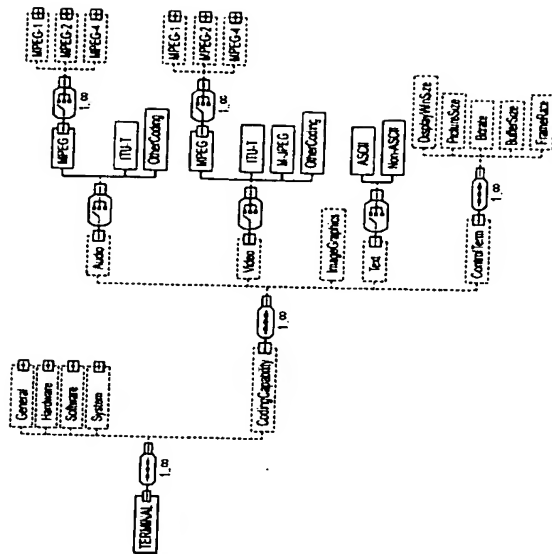


Fig 7

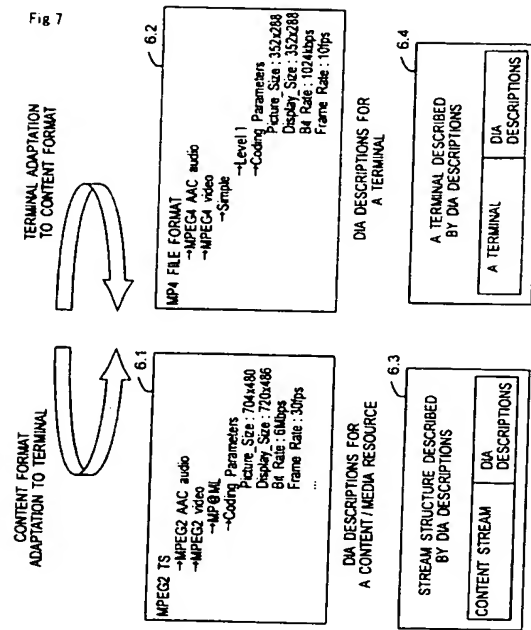


Fig 8

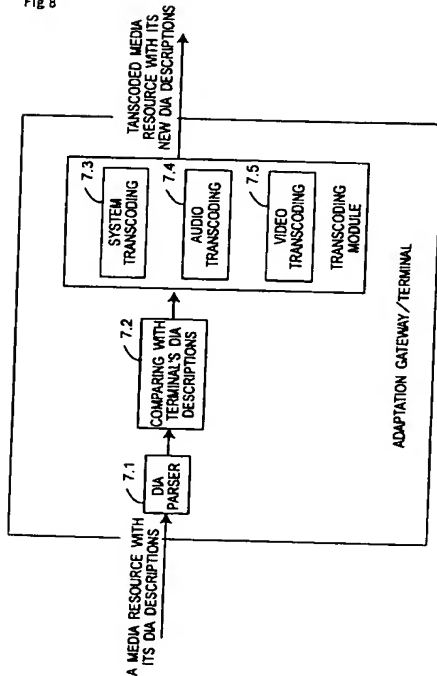


Fig 9

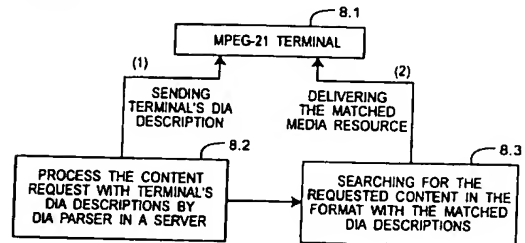


Fig 10

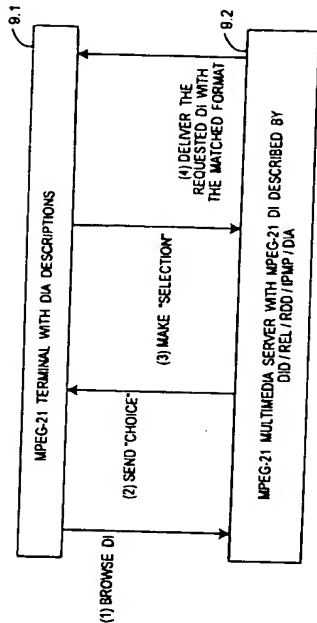


Fig 11

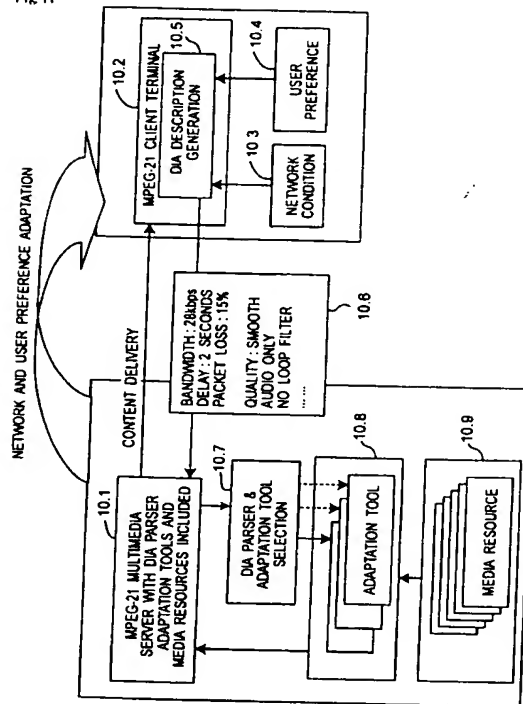


Fig 12

